

The University of Edinburgh



Department of Computer Science

Domain Theory in Realizability Toposes

by

Wesley Phoa

CST-82-91

(also published as ECS-LFCS-91-171)

July 1991

James Clerk Maxwell Building,
The King's Buildings,
Mayfield Road,
Edinburgh,
EH9 3JZ.

Domain Theory
in Realizability Toposes

Wesley Phoa
Trinity College, Cambridge

November 1990
Additions and revisions, July 1991

Abstract

This thesis describes some new categories of domains. These categories have two novel, and important, features:

- They are *full subcategories* of certain toposes. A topos can be regarded as a model of constructive set theory; so our domains are simply sets, with no extra structure. This makes it much simpler to describe constructions on domains: for instance, exponentials are ordinary set-theoretic function spaces.
- They are *small complete categories* (in an appropriate sense). This makes it simple to define polymorphic types; it also means that convex powerdomains, for example, exist for quite trivial reasons.

The use of category theory, and the internal language of a topos, makes most of the results quite easy to formulate and prove.

The first part of the thesis is a brief review of Hyland's theory of realizability toposes—in particular, the *effective topos*, the world of recursive mathematics. Inside each realizability topos is the small complete category of *modest sets*, the counterpart of the PER model. We can use it to model various polymorphic typed λ -calculi, including ones with subtyping.

In the next few chapters, we study domains in the effective topos. Building on work of Scott and Rosolini, we base the theory on the notion of *r.e. subobject*; we use it to define an intrinsic preorder on every object of the topos. Our predomains are the *complete Σ -spaces*: the objects for which this preorder is a chain-complete partial order. Every map between two complete Σ -spaces is automatically monotone and preserves suprema of chains; they form a complete full subcategory of the modest sets.

There is a natural notion of lifting for predomains, namely the 'computable partial map classifier'; this defines a monad on the category, whose algebras are the complete Σ -spaces with a bottom element—the domains. The categories of predomains and domains have many other attractive properties.

In the last couple of chapters, we find categories of domains in other realizability toposes: those based on the Plotkin/Scott graph models, and on the term model of the untyped λ -calculus in which two terms are identified if they have the same Böhm tree. The methods used are rather different, and the results here are still a little mysterious.

Statement

I hereby declare that, except where otherwise indicated, the results described in this thesis were obtained through my own unaided work.

I also declare that this dissertation is not substantially the same as any other that I have submitted for a degree or diploma at a university other than the University of Cambridge.

In this second version, chapters 3 and 4 have been added, and chapters 1, 12 and 13 have been revised. A number of other small changes have also been made.

Acknowledgements

This work was done under the patient supervision of Martin Hyland; I hope his influence is apparent throughout. For over two years he has been astoundingly kind and helpful. If I had followed all his advice I would have written a much better thesis.

Peter Johnstone and Andy Pitts were both generous with their time, and their guidance and support were invaluable. In particular, Andy Pitts sorted out a number of practical things for me, and Peter Johnstone helped rescue me from several nasty crises.

I also benefited from discussions with (among other people) Martin Abadi, Samson Abramsky, Roberto Amadio, Simon Ambler, Nick Benton, Roy Crole, Peter Freyd, Carl Gunter, Barney Hilken, Mike Johnson, Eugenio Moggi, Gordon Plotkin, Eike Ritter, Edmund Robinson, Paul Taylor, Dominic Verity, Fer-Jan de Vries and Glynn Winskel, and from discussions and correspondence with Pino Rosolini, Thomas Streicher, Jaap van Oosten and especially Dana Scott. Pino Rosolini kindly read an earlier draft of this thesis and made some useful criticisms. Nick Benton also kept a copy of an earlier manuscript in his luo.

Paul Taylor was actually my collaborator for a while (see section 9.1 and [59]); this seems like a good place to apologise to him for being such a difficult person to work with.

I was lucky to be given the opportunity to discuss all this material with many people at Århus, Edinburgh and Imperial College; I learnt a great deal at these places, and many things became clearer. I would especially like to thank Anders Kock and Jørn Schmidt, Mike Fourman and James Mckinna, and Steve Vickers and Paul Taylor for their kind invitations, and the stimulating talks I had with them.

Harold Simmons (running SERC's Logic for IT initiative) and the organizers of various PSSIs made it possible for me to meet lots of people, make lots of friends and learn a great deal more about mathematics and computer science than I could have on my own. I owe them an enormous debt.

I have received financial support from the Australian Committee of the Cambridge Commonwealth Trust through a Packer Scholarship, and from Trinity College through an Honorary Research Studentship. I have also received support

from the Department of Pure Mathematics and Mathematical Statistics, the Science and Engineering Research Council and the European Community, through the ESPRIT 'Categorical Logic in Computer Science' Basic Research Action. I am very grateful to all these bodies for their generosity.

Nearly all the new material in this revised version of my thesis came to light while I was working at Sydney University after submitting. I owe a lot to all the people there: Bob Walters, Terry Gagen, Mike Johnson, Sean Carmody and Giulio Katis especially. I also received some indirect aid from the Australian Research Council.

My thesis examiners were Peter Johnstone and Edmund Robinson; they had many useful suggestions, some of which I have tried to follow in revising this document.

Finally, I should thank all the people who've helped me in other ways through the years; to name them would take many pages. But there are four that are, for different reasons, special. My parents have always encouraged me, and made sacrifices for my sake. My friendship with Kristin English has sustained me for many years; I couldn't have coped without her support, and I can only regret the price she has had to pay. And Scott Johnson's help, at an important time in my life, has shaped my career ever since. I wish I could express my gratitude; all I can do now is dedicate this thesis to his memory.

Contents

1	Introduction	9
1.1	Backdrop	9
1.1.1	What do we want from a category of domains?	9
1.1.2	'Shortcomings' of classical domain theory	10
1.1.3	Three new approaches	11
1.1.4	The mathematical framework	12
1.1.5	How to read this thesis	13
1.2	Summary	13
1.2.1	Preliminaries	14
1.2.2	Domains in $\mathcal{E}ff$	15
1.2.3	On attempting a topological analysis	16
1.2.4	Looking for domains over other models	16
1.3	Some comments	18
1.3.1	An approximate history	18
1.3.2	Pros and cons of $\mathcal{E}ff$	20
1.3.3	PERs and the 'semantic scaffolding' viewpoint	21
1.3.4	PER analysis may be overspecification	22
1.3.5	On practical matters	23
2	Realizability toposes	24
2.1	PER models (a memory jogger)	24
2.2	Assemblies	27
2.2.1	Internal completeness properties	28
2.3	Building a topos	30
2.4	The internal language	32
2.4.1	On other realizability toposes	35
2.5	Modest sets	36

2.6	Completeness of modest sets	37
2.6.1	Completeness in terms of fibrations	38
2.7	Reflectivity	40
2.7.1	Reflectivity of subcategories	41
2.8	Relative computability	41
3	More on polymorphic types	46
3.1	Preliminary remarks	46
3.2	The groupoid interpretation	47
3.2.1	Parametricity	50
3.3	Uniformized interpretations	52
4	Fibrations and subtypes	55
4.1	Introduction	55
4.1.1	Introductory remarks	55
4.1.2	Fibrations and polymorphic λ -calculus	57
4.2	Modelling subtypes using fibrations	59
4.2.1	Rules for subtypes and bounded quantification	59
4.2.2	Characterizing inclusions	61
4.2.3	Incorporating bounded quantification	63
4.2.4	Incorporating subkinds	64
4.3	Examples	65
4.3.1	PER models	65
4.3.2	Models arising from PER models	67
4.3.3	Subtypes and constructions on types	69
5	Σ-spaces as predomains	71
5.1	The dominance Σ	71
5.2	Σ -spaces	72
5.3	Chain-complete posets	76
5.4	Linked Σ -spaces	77
5.4.1	When is the link relation transitive?	81
5.4.2	Sketch of a more external approach to completeness	82
5.5	Complete Σ -spaces	84
5.5.1	Note on the category $\mathcal{CSig}$	87
5.6	Flat Σ -spaces	89

6	Some miscellaneous results	91
6.1	Chain completeness and orthogonality	91
6.1.1	Not $\neg\neg$ -separated domains?	93
6.2	A limit-colimit coincidence	94
6.3	All functors are enriched	95
6.4	Lax limits	96
6.5	Weighted limits	98
7	Lifting and strictness	101
7.1	Focal Σ -spaces	101
7.1.1	On the fixed point category of $\mathcal{E}ff$	102
7.2	The lift construction	103
7.3	Computational interpretation of lifting	107
7.4	Strictness and the lift monad	110
7.5	Projection analysis	115
8	Coproducts and fibres	117
8.1	Preliminary remarks	117
8.2	Coproducts	118
8.3	Fibres	120
8.4	Projections	121
9	Algebraic constructions	123
9.1	The convex powerdomain	123
9.2	Modalities	126
9.3	Other constructions	127
9.4	Upper and lower powerdomains	129
10	Topological questions	131
10.1	The Σ -doctrine	131
10.2	Open spaces and maps	133
10.3	Topology, points and sobriety	136
10.4	Repletion	140
10.5	An example in two models	142

11 Sheaves	147
11.1 Definition and examples	147
11.2 Sections and stalks	149
11.2.1 A more concrete construction of stalks	151
11.2.2 Local homeomorphisms?	152
11.3 Continuously dependent types	152
11.3.1 A sheaf model for continuously dependent types	154
11.4 Specification spaces	157
11.4.1 Defining predicates	159
11.5 Different dominances	160
11.6 Gros toposes	160
11.6.1 So what?	162
11.6.2 A problem with the site Sig	163
11.6.3 Open surjections?	163
12 The graph model	165
12.1 Calculating indexed products	165
12.2 The r.e. graph model	166
12.2.1 Why look at the graph model?	167
12.2.2 Relative dominances	167
12.3 Partial maps and lifting	168
12.4 A partial order	170
12.5 Chain completeness and fixed points	172
12.5.1 Least fixed points	175
12.6 Combining them	176
13 The Böhm tree model	179
13.1 A criterion for reflectivity	179
13.2 Partial maps and lifting	180
13.3 A kind of chain completeness	183
13.4 All maps are stable?	185
13.5 Putting it all together	187
13.5.1 Closure under lifting	189
14 Related work and loose ends	192
14.1 Fixed point categories	192

14.2	The work of Rosolini	193
14.3	Extensional PERs	195
14.4	Domains in other models	196
14.4.1	Domains in Grothendieck toposes	198
14.5	Synthetic domain theory	199
14.5.1	The approach of Freyd	199
14.5.2	The approach of Hyland	199
14.6	Sound predicates	201
14.7	Final remarks	202

Chapter 1

Introduction

1.1 Backdrop

A modern functional programming language is likely to have a complex and powerful type discipline. To understand it and to prove things about it, we need a semantics which is mathematically precise but also simple, and which gives us some insight into the nature of the constructions that it models. In this thesis, we propose a new framework for denotational semantics, founded on category theory and constructive set theory. The basic ideas are simple but the results are surprisingly powerful.

This introduction is split into three sections. The first is a kind of ‘minimal’ introduction: it quickly explains the problems we want to solve, and the point of view we will take. Then, in 1.2, there is an overview of the thesis. Finally, the last section contains a short history, as well as some pretentious remarks of a philosophical nature. You may prefer to skip over them.

1.1.1 What do we want from a category of domains?

This is a good question to ask. It should be the motivation behind everything we do. (But we’d better not be too specific at this stage, because the real answer has to be, ‘whatever we can get’.)

Well, firstly—and this is what’s in the front of everyone’s mind—we want lots of structure so we can model lots of different programming language features. For example, we could ask for

- fixed points of endofunctions, to give us recursion;

- solutions of domain equations, so we can define recursive data types;
- powerdomains, to model nondeterminism;
- ways to model subtyping and inheritance;
- polymorphic types, ... and so on.

And if we're looking for a category of effective domains, we also want some formal analogue of the idea that 'all the maps we're considering are computable'.

But that isn't all. We also want a clean and well-motivated theory; after all, someone's going to end up using it. So:

- the constructions should be natural and intuitive
- there should be a simple and sensible treatment of computability
- we should not let the mathematics get too hairy

And these are things that a mathematician doing domain theory might be tempted to forget about.

[Another thing worth considering is that we might want to have different categories of domains for different purposes. But perhaps this is more questionable.]

1.1.2 'Shortcomings' of classical domain theory

There had better be something wrong with the way people traditionally do domain theory. Otherwise why start again from scratch? So this is a serious question—all the more serious because domain theory has been very successful.

But:

- The mathematics can get very complicated; there are dozens of different categories of domains, and the techniques developed for proving things about them—in fact, for defining them—have become very technical.
- Some features (for example, polymorphism) have proved to be quite tricky to model; the models are not quite satisfactory, or difficult to understand, or somewhat *ad hoc*, or sometimes all three.
- The treatment of effectivity (by enumerating bases and so on) causes lots of trouble and isn't very realistic anyway.

(I realize that I'm not being completely fair.)

So what's the problem? Perhaps that domain theory is based on continuous posets. This seems to have been motivated more by necessity than choice—it would have been nice just to use sets, but only cpos actually worked—though it's not too hard to explain, after the fact, why cpos capture our intuitions quite well.

But the analogy between computability and continuity is just an analogy, and it's bound not to be perfect. What's more, to graft effectivity onto all that via enumerated bases and so on looks artificial.

1.1.3 Three new approaches

Well, what else can we do? There are at least three directions we could follow. These are ideas that have been floating around for some time now; I won't give references here (but all three ideas can probably be traced back to Scott, though since then they have mutated considerably).

Axiomatic domain theory: We could simply try to axiomatize the properties a category of domains should have. This is not so simple: it's not even clear what sort of language we should use. And of course, we'd want to make sure our theory was consistent. But, if we could pull it off, it would be useful—we would have a kind of toolbox for language design and program specification, and no-one would have to worry about any concrete (and messy) categories of domains.

Domains as sets ('synthetic domain theory'): A primer on denotational semantics will tell you to 'think of a domain as a set'. Perhaps we can make this rigorous; perhaps we can think of a domain as a special kind of set in some constructive universe; this could well make sense if computability is built into the logic. And then constructions on domains would be set-theoretic, just the way we understand them, with no extra structure to worry about.

Domains as PERs: Start with some combinatory algebra A —think of it as an algebra of untyped primitive operations and data values—operations and objects that the *machine* understands. A data type or a semantic domain would be a (special kind of) *partial equivalence relation*, or *PER*, on A ; that is, to define a type X we would have to say

- what elements of A represent elements of X ; and

- given two different elements of A that represent elements of X , when to think of them as representing the *same* element of X .

And in the end, it would be nice if all three approaches were to tie up somehow. All three would fit into one unified framework, and we could flit from one point of view to another, depending on which happened to be most convenient at the time.

1.1.4 The mathematical framework

Regarding the first idea, one thing at least seems clear: we should try and use the concepts and methods of category theory. Things seem likely to work out more neatly that way; in fact, some ideas are almost impossible to express without using the language of categories.

As for the last two ideas: we've known for some time how to reconcile them. PERs on A turn out to be essentially the same thing as certain objects in a particular topos (=‘constructive universe of sets’)—namely the realizability topos on A .

This realizability interpretation of constructive set theory was first worked out by Powell (in an unpublished manuscript) and by Hyland in [27]. It will be more convenient to use Hyland's version; he uses the language of topos theory rather than set theory, which suits our category-theoretic approach.

The most familiar example is the *effective topos* $\mathcal{E}ff$; this is the realizability topos based on the partial recursive functions. Inside it, for example, every map $\mathbf{N} \rightarrow \mathbf{N}$ is a total recursive function (in a sense that can be made precise; see 2.4); and maps $\mathbf{N} \rightarrow 2$ correspond to recursive sets.

It turns out that the effective nature of the logic gives rise to an *intrinsic* partial order structure on certain objects; and all maps must respect this structure. In a way, it's transparent—we can use it to prove things, but if we want to ignore it, we can; we don't have to worry about defining a partial order on things, or about when it's preserved.

That's really the whole point: computability in the underlying model gives rise to intrinsic structure on (some of) the objects. We don't manipulate it—in fact we can't—we just observe it.

Technical note. To prove a result about $\mathcal{E}ff$, we will frequently give an informal constructive argument and then assert that ‘this can be carried out in the

internal logic of $\mathcal{E}ff$. My excuse is that a formal proof would be extremely long and unreadable, and would defeat the purpose of working in a topos-theoretic setting. Compare this with the use of ‘informal Church’s Thesis’ in recursion theory (see [67]) or the proofs of assertions that certain metamathematical arguments can be formalized in Peano Arithmetic.

1.1.5 How to read this thesis

Some parts of this thesis record work in progress, or ideas that haven’t worked out yet. Others are quite technical, and would probably only interest others people in the same field. Clearer and more definitive accounts will probably appear sometime in the future.

So if you are busy, or lazy, but for some reason feel obliged to read at least a part of this document, I would suggest looking just at the material on Σ -spaces: chapters 5, 6, 7, 8 and 9. (Chapters 6 and 8 could be skimmed.) The background material in chapter 2 is useful, but may make the subject more technical than it really is: messy details about the effective topos rarely appear in those later chapters.

As I’ve tried to indicate in the final chapter, other people working on the same kinds of problems have developed radically different points of view. I believe that this thesis is headed in the right direction; but I don’t advise you to take my word for it. Nor should you believe that chapter 14 gives a balanced or complete account; I couldn’t write one. Please read the other papers yourself.

1.2 Summary

This thesis is an account of work in progress. In fact, only the material on domains in the effective topos is in a fairly clean and satisfactory state; the rest may seem quite tentative, and a bit disjointed. I apologise for this—but I did think that there were results worth including, and other things worth saying, even though the big picture remains somewhat mysterious.

Rather than giving a detailed list of results, I’ve just tried to give an impression of what each chapter contains.

1.2.1 Preliminaries

Not too much background is assumed. On the mathematical side: some basic facts about category theory (including fibred categories) and the logical aspects of topos theory; and some non-technical recursion theory. Logic: some familiarity with constructive reasoning, and with the syntax and semantics of the λ -calculus. Computer science: a vague acquaintance with denotational semantics and the idea of polymorphism. The work described in this thesis is mostly independent of classical domain theory.

Chapter 2 briefly describes the mathematical setting in which we work: the theory of realisability toposes and modest sets. I've tried to motivate the subject by first discussing the PER model for polymorphism, and its 'internalisation' in the category of assemblies; it's natural then to try to construct a topos in which all these things live. There is also a discussion of the sense in which the category of modest sets is 'complete' and 'reflective'—notions which turn out to be surprisingly subtle. However, I haven't attempted to give a full account; that would fill a book. (The fact that no such book exists is a serious obstacle to further progress.)

The only original work in this chapter is contained in section 2.8; it has to do with relative computability. Though we'll refer to this section once or twice later on, it's not really of central importance. There are a few other observations which are 'folklore' (among the half a dozen people involved in the subject) but which don't seem to have appeared in print; for example, the comments on reflectivity.

The next two chapters do contain new results; they are not concerned specifically with domain theory, but all the results are applicable to the categories of domains discussed later on.

Chapter 3 discusses two new interpretations of polymorphism. The first, the *groupoid interpretation*, makes sense for any small complete category in a topos; it is meant to solve the problem of invariance under equivalence. The second, the *uniformized interpretation*, is applicable to reflective subcategories of modest sets (such as various categories of predomains). There are still many open questions concerning these interpretations.

In **chapter 4** we propose a general categorical framework for studying polymorphic typed λ -calculus with subtypes and bounded quantification; the idea is to impose a certain additional structure on a fibration that captures the notion of *inclusion*. Modest sets, and reflective subcategories of modest sets, provide ex-

amples of such structures; so we can integrate domain theory with subtyping and inheritance.

1.2.2 Domains in $\mathcal{E}ff$

Chapters 5 to 9 deal with domain theory in the effective topos $\mathcal{E}ff$. Most of this material is in a pretty satisfactory state.

In Chapter 5 we look for a category of predomains. We start by defining an object Σ in $\mathcal{E}ff$ which classifies ‘computable’ or ‘recursively enumerable’ subobjects. Such subobjects should be thought of as ‘Scott-opens’. The r.e. subobjects of an object give rise to an intrinsic preorder on it; we single out the objects for which this preorder is a chain complete partial order—such an object is called a *complete Σ -space*. These will be our predomains; they are necessarily modest sets. All maps between complete Σ -spaces are automatically monotone and preserve suprema of chains.

Any (internal) diagram of complete Σ -spaces has a limit in $\mathcal{M}od$; and it turns out that the Σ -order on the limit is the pointwise order. (The proof is not trivial.) So the category $\mathcal{C}Sig$ of complete Σ -spaces is a small complete category; more precisely, it’s as complete as $\mathcal{M}od$ is. Hence we can interpret polymorphic types. Another consequence is that $\mathcal{C}Sig$ is reflective.

Some other results about complete Σ -spaces that do not really fit into this chapter are gathered in Chapter 6. For example, we show that a limit-colimit coincidence holds for complete Σ -spaces; and we investigate their 2-categorical properties.

Chapter 7 deals with complete Σ -spaces that have a bottom element; these will be our domains. There is a natural lifted domain construction that classifies computable partial maps. The lift construction is actually a monad on $\mathcal{C}Sig$, the category of algebras for this monad is the category of complete Σ -spaces with a bottom element, and strict maps. This is therefore also a small complete category.

Since $\mathcal{C}Sig$ is reflective, we know that it has all small colimits; they are the reflections of the colimits in $\mathcal{E}ff$. But it would be nice if certain coproducts in $\mathcal{C}Sig$ were the set-theoretic ones. Chapter 8 contains some results in this direction.

In Chapter 9 we exploit the reflectivity of $\mathcal{C}Sig$ to perform some algebraic constructions. For example, there is a simple convex powerdomain construction on $\mathcal{C}Sig$: given a complete Σ -space X , form the object of finite (nonempty) subsets

of X and reflect back into $\mathcal{CSig}$ to get an object CX . The intrinsic order on CX is essentially the Egli-Milner order. The universal property on CX gives rise to modal operators which satisfy (versions of) the usual identities.

We can construct other kinds of domains with algebraic properties in a similar way; for example, domains that let us give a semantics for nondeterministic multi-valued computations. Furthermore, there are also upper and lower powerdomain constructions.

1.2.3 On attempting a topological analysis

One could take the view that the intrinsic order on an object X is really secondary. What comes first are the Σ -subsets of X ; thinking of these as a kind of ‘topology’ on X , one would like to develop domain theory along topological lines.

We look at this idea in **Chapter 10**; it turns out that there are difficulties. In particular, in section 10.5 we see that the ‘ Σ -topology’ on limits can be badly behaved. There’s also a short account of the (so far unsuccessful) search for a good category of ‘sober’ Σ -spaces, and a sketch of some related ideas of Hyland.

Chapter 11 deals with sheaves; unfortunately our poor understanding of the Σ -topology is a big problem. So this chapter just sketches ideas rather than presenting concrete results.

First we define the notion of *sheaf* on a Σ -space. After listing a couple of examples, we describe the sheaf of sections and stalk space functors, and a few other matters; and there is also a short discussion of gros toposes. I’ve chosen to define ‘sheaf’ in such a way that we can simply apply the theory already developed for ordinary topological spaces—this may turn out to have been a bad decision, but it’s a reasonable first attempt.

We also describe some possible applications of the notion of sheaf: to specifications, and also to making sense of the idea of ‘continuously dependent types’ in the context of complete Σ -spaces. It may be possible to regard sheaves as inclusive predicates as well.

1.2.4 Looking for domains over other models

It’s natural to wonder whether we can find domains in other realizability toposes. Chapters 12 and 13 deal with a graph model and a certain term model respectively; there are some concrete results, but a great deal of work remains to be done. The

problem is that the methods used in $\mathcal{E}\mathcal{F}\mathcal{F}$ don't seem to carry over; the notion of r.e. set doesn't generalize well to other models. In any case, it's interesting to find out whether other approaches will work.

Chapter 12 is about the r.e. graph model (but the same ideas work for other graph models). The situation here is, to a point, quite similar to that in $\mathcal{E}\mathcal{F}\mathcal{F}$ —though the techniques are rather different. We have an object Σ and a lift functor; we can find a small complete category whose objects are partially ordered (by a certain 'link relation': cf. 5.4) and have suprema of chains; among them, our 'domains' are the objects with a certain sort of bottom element. But the results seem a bit *ad hoc*, and rely to some extent on calculation with PERs.

The point here, perhaps, is that you don't need to use the Σ -order to do domain theory; it's possible to obtain a good order in another way. Of course, the results are not quite so satisfactory.

In **Chapter 13** we deal with the closed term model for the λ -calculus in which two terms with the same Böhm tree are identified (again, the same ideas work for a larger class of models). Again there is an object Σ , corresponding to a notion of computable partial map. But this time it's not clear how to define an intrinsic partial order! Still, using ideas of Hyland, we can isolate the objects with suprema of certain chains—the only ones we need to compute fixed points—and these form a small complete category. It's also possible to define a notion of 'meet-closure', such that all maps between meet-closed PERs are stable. As before, the results are still rather crude.

The pleasant thing about this model is that looking at the underlying PERs, fixed points are actually being computed by fixed point combinators—what's going on in the domains is closely related to the real computations in the untyped model. (In the theory in $\mathcal{E}\mathcal{F}\mathcal{F}$, such details are hidden by the layers of abstraction.) It seems likely, though, that the results proved in chapters 12 and 13 rely only on a few details of those models, and could be established for more general classes of models.

Incidentally, an odd feature of the results in chapter 13 is that they don't seem to be at all related to the axiomatic framework proposed by Hyland in [30]. There are quite a few other results in the field, due to others, which have no good abstract explanation.

In the last chapter, there's a very brief survey on other attempts to do domain theory 'synthetically', and on the search for domains inside categories of PERs

or inside realizability toposes. Then there's a vague remark about one way to enrich type theory to make it more appropriate for talking about domains (by adding extensions of certain predicates); and last of all, some final remarks of an undemanding nature.

1.3 Some comments

I want to give a little historical background, and then make some remarks about the general approach taken in this thesis—or, to put it more pompously, the underlying philosophy. But before I do that, there are some features of this thesis that I should try to justify.

Firstly, all this looks like mathematics. And in fact, the things that we'll be studying are (in my opinion) quite interesting in their own right. But this is secondary to the main purpose, which is to develop tools—more than that, to develop a conceptual framework—for the use of computer scientists.

So there are lots of results, and some examples, and some discussion, but no climactic theorems. In fact, nothing is even labelled a 'theorem' (well, 2.8.1 is, but only because it's not useful). And though we may sometimes seem to be a long way away from computer science, it should always be in the back of our minds.

Secondly, I've gone against the advice of several people by concentrating on some particular models—some specific realizability toposes—instead of trying to develop a general theory. In a way this is more a matter of attitude than practice; after all, even when working inside one model, the easiest techniques to discover and apply are usually the most abstract.

The point, though, is that—at least for me—insight comes from examples rather than from heaven. And perhaps there will be no single 'synthetic domain theory', but many theories, or even many languages, for different purposes. One way to find out is to study lots of examples; bearing in mind that we will, in the end, want to free ourselves from the messy details of specific models. That is, we should try to *study concrete things in an abstract way*.

1.3.1 An approximate history

Domain theory began with the discovery of the D_∞ and graph models for the untyped λ -calculus. This work gave rise to the idea of modelling types/semantic

domains by using complete partial orders or certain topological spaces (see, for example, [73]); operations or programs are then continuous functions.

Some time later, two new ideas emerged, beginning with a talk by Scott at the 1979 Peripatetic Seminar on Sheaves and Logic in Sussex:

1. Domain theory should be a synthetic theory, based on suitable axioms.
2. A domain should be thought of as just a special sort of set, so that domain-theoretic constructions are set-theoretic in nature—in an appropriate constructive universe.

His specific suggestion was to look for toposes with nontrivial fixed point categories (see 14.1). The objects with the fixed point property would then provide a suitable cartesian closed category of domains.

Scott later showed that categories of domains could be embedded in presheaf toposes via the Yoneda embedding, in such a way as to preserve the cartesian closed structure—so the ‘domains as sets’ idea is at least consistent. Of course, from a conceptual point of view, this approach is back-to-front: the topos (set theory) should come first, *then* the domains.

Further progress had to wait for developments in topos theory. Following on from initial work by Powell and Hyland on realizability and set theory, McCarty showed in [46] that (from an external point of view) classical categories of effective domains could be embedded in the effective topos. And Hyland and Rosolini showed, by somewhat *ad hoc* methods, that the effective topos and Mulry’s recursive topos both contained nontrivial fixed point categories.

This last result grew out of work on the r.e. subobject classifier. The basic ideas go back to Mulry; it was Scott who saw how to give an internal definition of Σ ; and in his thesis, Rosolini developed a coherent theory of r.e. subobjects and used it to build (still in a fairly classical way) a category of domains in $\mathcal{E}ff$ whose definition was entirely internal. See 14.2.

In another direction, Moggi noticed that PER models of the second order polymorphic λ -calculus should correspond to internal, complete full subcategories of realizability toposes—and in fact, Moggi and Hyland then proved that $\mathcal{E}ff$ contained a small complete category, essentially the category of PERs, thus making possible a ‘set-theoretic’ interpretation of polymorphism: [28], [45]. Incidentally, it wasn’t a simple matter even to find the correct notion of ‘small complete category’: cf. [33], [65].

It then became natural to look for small complete categories of (pre)domains inside $\mathcal{E}ff$. (It was hoped that Rosolini's category of σ -sets would be reflective, but this conjecture has so far resisted proof.)

[Much of the preceding account was lifted from a talk by Martin Hyland: see [29]. It is probably not very accurate; I'm not sure if anyone could reconstruct the history of these ideas.]

1.3.2 Pros and cons of $\mathcal{E}ff$

The effective topos is the realizability topos corresponding to Kleene's first model, i.e. indices for the partial recursive functions. So the notion of computability it embodies is the 'canonical' one, accepted by anyone who believes Church's Thesis—and it contains the world of recursive mathematics as we ordinarily understand it (though it contains a lot more besides).

So—though I don't want to press this point—results about $\mathcal{E}ff$ have a certain philosophical significance. For example, inside it we would expect to find 'the domain theory corresponding to our intuitive notion of computation'. We might expect it to play a central role in any conceptual investigation of synthetic (effective) domain theory; though we might still, sometimes, choose to use domains in other toposes.

There is a more pragmatic reason for working inside $\mathcal{E}ff$ rather than other realizability toposes. The internal logic of $\mathcal{E}ff$, and the principles that hold within it, are relatively well understood; well enough so that it's possible to develop our arguments 'just as if we were doing ordinary recursive mathematics'. This leads to a theory that is intuitive, high-level and (hopefully) accessible.¹ We can forget about the underlying model.

But because we can, we do. And perhaps there are phenomena which this approach obscures: axiomatization means hiding information, which can be a disadvantage as well as an advantage. There may be things going on in the black box which we're actually interested in.

[Perhaps this is illustrated by the following remark, due to Martin Hyland: we don't know how the second recursion theorem fits into the internal logic of $\mathcal{E}ff$; we don't even know how to talk about it.]

¹You should compare the results proved inside $\mathcal{E}ff$ with the ones obtained for other realizability toposes. See section 2.4 for an explanation.

Another problem: the partial recursive functions have nothing to do with how computations are carried out in practice. If we want information about the relationship between denotational and operational semantics, *Eff* may not be a good place to do our denotational semantics. Realizability toposes based on term models may be more realistic. Let's discuss this further under a new heading.

1.3.3 PERs and the 'semantic scaffolding' viewpoint

Recall from section 1.1 the idea of 'domains as PERs'. What's the justification for taking this point of view? (Apart from the fact that it works.)

We can think of a combinatory algebra as a sort of abstract machine. It consists of untyped operations and data values; it encapsulates a particular notion of computation. For example, a term model for the λ -calculus and a term model for the calculus of combinators both describe, in different ways, a notion of computation that works on concrete expressions. One could imagine implementing a term model on an actual machine (and indeed there are machines whose 'instruction sets' are based on combinatory logic; and implementations of functional languages are often based on the λ -calculus or on combinators.)

If we say that domains are PERs, we're really taking the view that the type discipline is purely for the programmer's convenience; it's just semantic scaffolding that's thrown away when the program's compiled; at run time it's all back down to untyped elements of A acting on each other. In other words, we don't really work with 'real' domains X whose members have codes that are elements of A ; rather, the programmer is choosing to regard particular elements of A , for some purposes, as representing members of a 'type' X that is part of the language, not something the machine cares about.

[This is closely related to the point of view advocated by Hindley, who suggests that type inference systems for the untyped λ -calculus are more important, and more fundamental, than the type λ -calculus. But one problem with the categories of domains discussed in this thesis is that we do not have corresponding type inference rules for them yet. Note that it really only makes sense to ask for them when we are dealing with term models—that is, in the context of chapters 12 and 13.]

But if we choose to think this way, then developing a domain theory over these models is going to be different from working in *Eff*. This time we care about the

details of the underlying model; we want to investigate what effect they have on the structure and behaviour of our domains. For example, what do fixed points in our domains have to do with fixed point combinators? (This is not a silly question; in the functional language PONDER, recursion is actually implemented using a fixed point combinator. Actually, I suppose it *is* a silly question if you think PONDER is a silly language.)

This doesn't mean we're going to try and make all our proofs dirty. On the contrary, it'll be helpful to try and make things as abstract as possible without losing sight of what's happening in A . And it would be silly not to try and abstract, eventually, away from the specific combinatory algebra A ; to find general features of A that give rise to particular sorts of domains.

1.3.4 PER analysis may be overspecification

There's a limitation of the 'domains as PERs' viewpoint. It's this: we begin with a primitive algebra of (untyped) operations and values, and *then* construct a category of domains. That is, a notion of implementation comes first.

[This is not quite right; picking a model doesn't oblige us to say anything about reduction. Perhaps it's better to put it this way: we have to specify some static details about the machine—its instruction set, the data values it recognizes— but we don't have to specify the dynamic process by which it computes things.]

So if we use these domains to give a denotational semantics for a language, we are committing ourselves to particular details about its implementation.

[One could say the same thing about using domains in *Eff*. But the point is, when we work in *Eff* the underlying model drops out of sight; the whole point of looking at these other models is that we get categories of domains *plus* a very PER-oriented way of looking at them.]

Of course, in an ideal world we would have a 'generic' synthetic domain theory together with information about how these 'abstract domains' were related to concrete categories of domains in various realizability toposes; or, more flexibly, an all-purpose language of synthetic domain theory, together with various theories corresponding to different models, perhaps with a common core.

1.3.5 On practical matters

In this thesis I have made no attempt to model particular languages or formal systems. I've simply tried to explain the mathematical ideas; and I think they're best understood when not obscured by reams of syntax.

Of course, the ideas can only be tested by applying them; but we should not allow existing languages to hamper our freedom of movement. For example, it is not a disaster if we can't model the language Quest precisely as it stands: this, and other systems, are simply research tools. The point is to try to model general features of languages, and to let the mathematics and the formal systems evolve alongside each other.

Actually, the practical aspects of the mathematics are sometimes quite unintentional. For example, the results in section 8.3 were proved purely out of curiosity; I was unaware of the work of John Launchbury, which explains why such results are useful in dealing with partial evaluation. And similarly, section 7.5, which shows how quite abstract results can be used to talk about the compile-time analysis of functional programs in our context, was only added after I went to a talk by Phil Wadler.

These are only small examples, but they do seem to illustrate the odd relationship between mathematics and its applications.

Chapter 2

Realizability toposes

2.1 PER models (a memory jogger)

One of the most important ideas in this thesis is that we should try to construct small complete categories, and then exploit the fact that they are small and complete. In this chapter, we see how to construct small complete categories by considering models of computation. What's more, we'll see how they arise as categories of *sets*; that is, as small full subcategories of certain toposes, called *realizability toposes*.

We begin this chapter by giving a very sketchy description of the PER model for polymorphism—this will turn out to be the ‘external’ counterpart of a small category in a given realizability topos $\mathcal{E}$. So in this section and the next, we describe the situation from an external point of view; and in the rest of the chapter, we see what everything looks like from inside $\mathcal{E}$, using the internal language of $\mathcal{E}$. This is less hands-on, more abstract, but more powerful too.

The idea behind the PER model (for finite types) goes back to the HEOs and Kreisel: [43]. It was Girard, though, who first noticed that PERs could be used to model polymorphism: [22]. In recent years, PER models have been widely studied, mostly by computer scientists, and mostly externally, without using topos theory. It would be tedious to try and list every reference; [45] is a good place to start, though.

Let A be a partial combinatory algebra (see [9]); that is, A is a set equipped with a partial binary operation—‘application’—and certain distinguished elements k and s satisfying certain identities. (Note that we'll be using *lower case* letters

for combinators.) Given these, we can define lots of other things, such as pairing and projection combinators. These will be used without comment.¹

The three examples we'll be looking at later are:

- $K_1 (= \mathbb{N})$ indices (Gödel numbers) for partial recursive functions
- R the r.e. graph model
- B a particular term model of the λ -calculus.

Of these, the first—called *Kleene's first model*—is the most important to us; the corresponding realizability topos is called the *effective topos*.

The other two are actually total combinatory algebras; see [9] or [8] for the definitions, and more information. For the moment, it's best just to keep the first example in mind.

Recall that a **partial equivalence relation** or **PER** is a relation which is symmetric and transitive, but not necessarily reflexive. (Observe that the collection of partial equivalence relations on a set is closed under arbitrary intersection.)

Let Per be the set of partial equivalence relations on A . We can make Per into a category as follows: if R, S are PERs then a map $R \rightarrow S$ is a function

$$f: A/R \rightarrow A/S$$

between the sets of equivalence classes, for which there exists an element $a \in A$ such that for all $p \in A$,

$$pRp \longrightarrow [ap]_S = f([p]_R)$$

where $[p]_R$ is the R -equivalence class containing p . (That is, f is a function taking R -equivalence classes to S -equivalence classes that is 'tracked' by an element of A .)

For example, if $A = \mathbb{N}$ (indices for partial recursive functions) and R and S are both the identity relation then a map $R \rightarrow S$ is just a map

$$f: \mathbb{N} \rightarrow \mathbb{N}$$

¹Actually, there is an issue here that we are glossing over. Namely, if we need an object of A that does a particular thing, it's not quite enough to write down an appropriate λ -expression; A is only a *partial* combinatory algebra, so it is not immediately obvious that this expression, when translated into a combinator expression so as to yield an element of A , will do what it is intended to. In fact, whether it does might even depend on the translation. This observation is due to Luke Ong; he solved the problem by proving a lemma about the interpretation of λ -expressions in (quasi) partial combinatory algebras. Details will appear in a forthcoming paper by Hyland and Ong.

for which there exists an index $e \in \mathbb{N}$ such that $f(n) = \{e\}(n)$ for all $n \in \mathbb{N}$; that is, it's just a total recursive function.

Proposition 2.1.1 *Per is a cartesian closed category with finite limits.*

For instance, if R, S are PERs then their exponential S^R may be defined by putting

$$eS^R f \longleftrightarrow (\forall p, q \in A. pRq \rightarrow (ep)S(fq)).$$

It therefore provides us with a model for the simply typed λ -calculus: just interpret product types by categorical products, and function space types by exponentials. In the case $A = \mathbb{N}$, the finite type structure we get is just the hereditarily effective operations (that is, the *extensional* things, not the hereditarily recursive operations); cf. [27].

The interesting thing, however, is that Per provides a model for various *polymorphic* λ -calculi.

Consider the second-order typed λ -calculus of Girard and Reynolds (see, for example, [23]). Its main feature is that given a type expression $T\alpha$ with free type variable α , we can form the '*polymorphic*' type

$$\Pi\alpha.T\alpha$$

(with corresponding term-forming operations and rules); and, given a term of type $\Pi\alpha.T\alpha$ we can extract, for any particular type X , a term of type TX . (This is called *instantiation at X* .)

Intuitively, an element of the type $\Pi\alpha.T\alpha$ is 'something which is uniformly an element of all types TX ' (clearly an impredicative concept). For instance, an element of $\Pi\alpha.\alpha \rightarrow \alpha$ should be something that, given any type X , yields a map $X \rightarrow X$; an example would be the *polymorphic identity*, which instantiates to the identity function at every type.

In the PER model we can interpret polymorphic types by intersection: loosely speaking, the PER corresponding to $\Pi\alpha.T(\alpha)$ is the PER

$$\bigcap_{R \in \text{Per}} T(R).$$

For example, the interpretation of $\Pi\alpha.\alpha \rightarrow \alpha$ is

$$\bigcap_{R \in \text{Per}} R^R$$

and the identity combinator $i = \text{skk} \in A$ is indeed in the domain of this PER. (In fact, for some PER models such as K_1 , or open term models for the λ -calculus, it's the only element in the domain of this PER: see [34].)

We'd like to think of $\bigcap_{R \in \text{Per}} T(R)$ as the product (indexed by the set of PERs) of all the PERs $T(R)$. Unfortunately, it just isn't the categorical product in Per ; it only has a universal property with respect to 'uniformly given' families of maps. To make sense of this remark, we have to be more subtle; but, as you'll see, it does have a natural meaning.

2.2 Assemblies

There are more sophisticated syntactic theories of polymorphism: for example, the *theory of constructions* of Coquand and Huet (see [14]). To show that PERs can also model these, we need some more machinery. (See also chapter 4 for a discussion of weaker systems like $F\omega$.)

Again, let A be a partial combinatory algebra. An A -valued assembly (or just an assembly if A is understood) consists of a set X together with, for each $x \in X$, a nonempty subset $|x \in X| \subseteq A$. We think of this as a set of 'codes for x '; as x varies, these subsets need not necessarily be disjoint.

By abuse of notation, we usually let X denote the assembly, and not just the underlying set. Also, it's worth noting that in the case $A = K_1$, A -valued assemblies are usually called ω -sets.

If X and Y are two assemblies, a map $X \rightarrow Y$ is defined to be a function $f: X \rightarrow Y$ between the underlying sets, for which there exists an element $a \in A$ such that for all $x \in X$,

$$p \in |x \in X| \longrightarrow ap \in |f(x) \in Y|$$

(we say that a tracks f). Let Sep be the category of (A -valued) assemblies; this seemingly odd notation will be explained in the next section.

Proposition 2.2.1 *Sep has finite limits and is cartesian closed.*

For example, if X and Y are assemblies then their exponential Y^X may be defined by letting

- underlying set of $Y^X = \{\text{maps } f: X \rightarrow Y \text{ in } \text{Sep}\};$

- $|f \in Y^X| = \{a \in A \mid a \text{ tracks } f\}$.

(For more details see [12]).

Sep contains the ordinary category of sets: if X is a set, denote by ∇X the assembly defined by:

- underlying set is X ;
- for $x \in X$, $|x \in X| = A$.

It's easy to see that any set function $X \rightarrow Y$ gives rise to a map of assemblies $\nabla X \rightarrow \nabla Y$ (tracked by, for example, k ; or, if A is total, any element of A). So we get a full embedding $Sets \hookrightarrow Sep$. This even has a left adjoint: see the next section.

Call an assembly a **modest set** if two different elements never share a common code; that is, if for all $x, x' \in X$,

$$x \neq x' \longrightarrow |x \in X| \cap |x' \in X| = \emptyset.$$

Let *Mod* be the full subcategory of *Sep* whose objects are the modest sets.

We have seen this category before. For notice that any PER R gives rise to an assembly:

- underlying set is $A/R = \{\text{set of } R\text{-equivalence classes}\}$;
- if $u \in A/R$, $|u \in A/R| = u \subseteq A$.

(That is, the codes for an equivalence class are the elements of the equivalence class.) Then we get another full embedding $Per \hookrightarrow Sep$ that preserves finite limits and the cartesian closed structure; and the image of I is equivalent to *Mod*.

The following result will come in handy.

Lemma 2.2.2 *Let X be an assembly. There is a natural bijection between elements of the underlying set of X and global sections $1 \rightarrow X \in Sep$.*

2.2.1 Internal completeness properties

Since *Sep* has pullbacks, it is fibred over itself in the usual way. We think of an object $(\begin{smallmatrix} X \\ I \end{smallmatrix}) \in Sep/I$ as an I -indexed family $(X_i)_{i \in I}$ of assemblies; then a map $(\begin{smallmatrix} X \\ I \end{smallmatrix}) \rightarrow (\begin{smallmatrix} Y \\ I \end{smallmatrix})$ is a family of maps $(X_i \rightarrow Y_i)_{i \in I}$ tracked by a single code 'uniformly in I ' (it comes from a map $X \rightarrow Y$, remember).

Proposition 2.2.3 *Sep is locally cartesian closed; in fact, it's a quasitopos.*

This can be proved by some laborious computations, but it will be obvious once you read the next section.

(The result implies that *Sep* has indexed products relative to itself; that is, in the fibration, pullback functors have right adjoints that satisfy the Beck condition.) However, *Sep* is *not* a topos—it does not have good quotients of equivalence relations.

Let $\mathcal{M}od(I)$ be the full subcategory of $\mathcal{S}ep/I$ consisting of those families $(\overset{X}{\underset{I}{p}})$ for which each $X_i = p^{-1}(i)$ is modest. This defines a subfibration of *Sep* fibred over itself, which we also denote by $\mathcal{M}od$. It is complete relative to *Sep* in the following sense.

Proposition 2.2.4 (i) *Each $\mathcal{M}od(I)$ is closed under finite limits in $\mathcal{S}ep/I$.*
(ii) *If $(\overset{X}{\underset{I}{p}}) \in \mathcal{M}od(I)$ and $\alpha: I \rightarrow J$ then $\Pi_\alpha(\overset{X}{\underset{I}{p}}) \in \mathcal{M}od(J)$; that is, $\mathcal{M}od$ is closed under indexed products.*

Informally, (ii) says that $\mathcal{M}od$ is closed under products indexed by an assembly. In particular, it is closed under products indexed by all ordinary sets, since $\mathcal{S}ets \hookrightarrow \mathcal{S}ep$. Notice that these products only have a universal property with respect to ‘uniform’ families of maps, so this has nothing to do with the fact that *Per* is *not* complete when considered as an ordinary category.

Now here’s the interesting part. Define an internal category $\mathbf{Mod}$ in *Sep* by

$$\begin{aligned} \mathbf{Mod}_0 &= \nabla\{\text{objects of } \mathbf{Per}\} \\ \mathbf{Mod}_1 &: \begin{array}{l} \text{underlying set} = \{\text{arrows of } \mathbf{Per}\} \\ |f \in \mathbf{Mod}_1| = \{\text{codes for } f\} \end{array} \end{aligned}$$

$\mathbf{Mod}$ is then an internal full subcategory of *Sep* whose externalization is the fibration $\mathcal{M}od$ just defined:

$$\mathcal{M}od(I) \simeq \mathcal{S}ep(I, \mathbf{Mod})$$

(this is all made precise, and justified, in [28]). Notice that $\mathbf{Mod}$ is *not* the image of the category $\mathbf{Per} \in \mathcal{S}ets$ under the functor ∇ : the object-of-arrows is different.

The point is that since modest sets are closed under limits indexed by assemblies, they are closed under products indexed by $\mathbf{Mod}_0$ —that is, we can interpret

polymorphic types as products, but ‘relative to *Sep*’ (an idea which can be traced back to Moggi, I think). This explains the comment at the end of the last section.

What’s more, the modest sets are closed under all sorts of other exciting kinds of products. In fact we get a model for the theory of constructions (see [32]) in which the kinds are assemblies and the types are (internal families of) modest sets. As these things get a bit complicated, and aren’t really relevant to what follows, we won’t say more; see [33] if you’re curious.

2.3 Building a topos

This is all very well, but calculations with assemblies can get hairy—especially when we have to work in general fibres Sep/I , which is necessary since the useful results are the ones which are valid in an ‘indexed’ or ‘parametrized’ sense.

It would be simpler if we could embed everything in a topos $\mathcal{E}$. Then we could talk about completeness, and so on, in the internal language of $\mathcal{E}$; so everything becomes (constructive) set theory, and we can forget about all our coding and references to the underlying partial combinatory algebra A . Of course, as we’ll see, working inside a topos has its own problems: things are less concrete and constructive reasoning can be very subtle.

The appropriate topos is the *realizability topos* on A . This theory begins with the work of Powell and Hyland on realizability and set theory; but there are now several construction of this topos. For example, one can

- (Hyland, Powell) Define the category of sets equipped with an A -valued equality relation (note that we could think of an assembly as a set equipped with an A -valued membership relation);
- (Freyd *et al.*) Add good quotients of equivalence relations to Sep by splitting certain idempotents in the category of relations;
- (Robinson and Rosolini) Freely adjoin recursively indexed coproducts to the category of sets...

In each case the proof that what you get is a topos is quite long and complicated. (Perhaps the best reference is [31]). So here we just give a very brief sketch of the construction—mostly so it can be compared with the definition of assembly. Be warned, then, that the following remarks are oversimplified.

Again, fix a partial combinatory algebra A . An A -valued set is a set X together with, for each $x, x' \in X$, a subset

$$|x = x'| \subseteq A$$

(an ' A -valued equality relation'; note that we can now define an A -valued membership relation $|x \in X| = |x = x|$). This 'relation' must be 'realizably symmetric and transitive'; that is, there must be elements $s, t \in A$ such that for all $x, x' \in X$,

$$p \in |x = x'| \longrightarrow sp \in |x' = x|$$

and for all $x, x', x'' \in X$,

$$p \in |x = x'| \wedge q \in |x' = x''| \longrightarrow t\langle p, q \rangle \in |x = x''|$$

(where $\langle \cdot, \cdot \rangle: A \times A \rightarrow A$ is a pairing map, definable by an element of A). In other words, there must be an operation taking a 'witness' for $x = x'$ to a witness for $x' = x$, an operation given by an actual element of A , uniformly in x and x' ; and similarly for transitivity.

Again we just denote this A -valued set by X . A map $f: X \rightarrow Y$ between A -valued sets X and Y is an ' A -valued functional relation' F : that is, for each $x \in X$ and $y \in Y$ a subset

$$F(x, y) \subseteq A$$

(which we could denote by $|f(x) = y| \subseteq A$) that is 'realizably a single-valued and total relation'. For instance, to say that F is realizably total means that there is an element $t \in A$ such that for all $x \in X$,

$$\begin{aligned} &\text{if } p \in |x \in X| \text{ then } tp = \langle q, r \rangle, \\ &\text{where } q \in |y \in Y| \text{ for some } y \in Y, \\ &\text{and } r \in F(x, y). \end{aligned}$$

(i.e. given a code for x , it provides a code for some y and a witness for the fact that $f(x) = y$).

Proposition 2.3.1 *The category $\mathcal{E}$ of A -valued sets is a topos.*

Any A -valued assembly X gives rise to an A -valued set: put

$$|x = x'| = \begin{cases} |x \in X| & x = x' \\ \emptyset & \text{else} \end{cases}$$

and one can extend this to maps to give a full embedding $\text{Sep} \hookrightarrow \mathcal{E}$. And by composing, we also get full embeddings $\text{Sets} \hookrightarrow \mathcal{E}$, $\text{Mod} \hookrightarrow \mathcal{E}$.

Proposition 2.3.2 (Hyland) *Consider the double negation topology on $\mathcal{E}$.*

(i) Sep is equivalent to the category of $\neg\neg$ -separated objects.

(ii) Sets is equivalent to the category of $\neg\neg$ -sheaves.

So we can study assemblies (and modest sets) using topos theory; and think of them just as ‘ordinary sets’ in a particular constructive universe. That’s one of the central themes of this thesis.

See [27] for more details about everything discussed in this section, and the next.

Note. For some time it was not known when different combinatory algebras gave rise to inequivalent realizability toposes; for example, Freyd conjectured that the effective topos was equivalent to the topos arising from closed λ -terms up to β -equality (presumably because both give rise to the same notion of computable function). Finally, Johnstone and Robinson showed in [39] that the effective topos is not equivalent to any realizability topos constructed from a total combinatory algebra.

However, the general problem of how to relate different realizability toposes remains open. There are some interesting early results of Hyland, as well as certain universal properties of $\mathcal{E}\text{ff}$ discovered by Pitts and (another) by Freyd, but a systematic classification is missing. So let me just mention that the three toposes studied in this thesis are not equivalent.

Carboni has asked whether it is possible to give an intrinsic characterization of realizability toposes (in the way that we can characterize spatial toposes, presheaf toposes and so on without reference to the underlying site). I don’t know of any work at all in this direction.

2.4 The internal language

We can make A into an A -valued assembly: put

$$|a \in A| = \{a\}.$$

And this gives us a $\neg\neg$ -separated object $A \in \mathcal{E}$.

In the case $A = K_1$, we call $\mathcal{E}$ the **effective topos** and denote it by $\mathcal{E}\text{ff}$. The object we just defined is denoted by $\mathbb{N}$ —its underlying set is the natural numbers and

$$|n \in \mathbb{N}| = \{n\};$$

and it turns out to be the natural numbers object in $\mathcal{E}ff$.

What's more, since the definition of $\mathcal{E}ff$ was inspired by Kleene's original realizability interpretation of intuitionistic logic, it's no surprise that $\mathcal{E}ff$ turns out to be 'the topos of recursive mathematics': the mathematical theorems valid in $\mathcal{E}ff$ are, roughly speaking, just the theorems of Russian constructivism (see [11]).

We briefly explain why this is so by listing some formulas (mostly involving $\mathbb{N}$) that are true in the internal logic of $\mathcal{E}ff$. Corresponding versions are, however, true in any realizability topos.

These formulas correspond to 'axioms' of recursive mathematics (so far as it has any). In the sequel, we'll exploit this as follows: to show that a certain statement is valid in $\mathcal{E}ff$, we will give an informal proof using these mathematical principles, and then assert that everything can be formalized to yield an argument valid in the internal logic of $\mathcal{E}ff$.

[In fact, most of the results in the first couple of chapters could be proved in any topos in which the following principles are valid; this kind of axiomatic approach has been advocated by Scott—see [72]. However, $\mathcal{E}ff$ seems to be the only suitable model (though lots can be done in Mulry's recursive topos).]

First we have **Church's thesis**:

$$\forall f \in \mathbb{N}^{\mathbb{N}}. \exists e \in \mathbb{N}. (\forall n. \exists y. T(e, n, y) \wedge U(y) = f(n)).$$

(Here T is Kleene's T -predicate, and U is Kleene's output function.) This says that every map $\mathbb{N} \rightarrow \mathbb{N}$ is a total recursive function. This should be plausible, remembering that the underlying set of $\mathbb{N}^{\mathbb{N}}$ is the set of total recursive functions.

Next, **Markov's principle**:

$$\forall \phi \in 2^{\mathbb{N}}. \neg\neg(\exists n. \phi(n)) \rightarrow (\exists n. \phi(n)).$$

Remember that $2 = 1 + 1$ can be thought of as the object of decidable truth-values $\{\top, \perp\} \rightarrow \Omega$. This statement then says that given a decidable property of natural numbers, which can't fail for every number, we can find a number for which it holds (intuitively, by testing each number in turn).

Finally, the **$\mathbb{N}$ -axiom of choice**:

$$\forall \phi \in \Omega^{\mathbb{N} \times X}. (\forall n. \exists x \in X. \phi(n, x)) \rightarrow (\exists f \in X^{\mathbb{N}}. \forall x \in X. \phi(x, f(x))).$$

In fact, the axiom of dependent choices also holds.—All the proofs, and much more information, appears in [27].

This is a picture of the world of mathematics as seen from inside $\mathcal{E}ff$. But we'll also find it handy to be able to relate the logic of $\mathcal{E}ff$ to the 'real world' of $\mathbf{Sets}$. This is also done in [27]; the point is that the fact that $\mathbf{Sets}$ is the topos of $\neg\neg$ -sheaves has useful consequences.

The basic fact we need is that if $X \in \mathcal{E}ff$, subsets of the underlying set of X correspond to $\neg\neg$ -closed subobjects of X . Let's see how this works in case X is $\neg\neg$ -separated (i.e. comes from an assembly).

Let $X \in \mathbf{Sep}$ and let $X' \subseteq X$ be a subset of the underlying set of X . Make X' into an assembly by putting, for all $x \in X'$,

$$|x \in X'| = |x \in X|.$$

Then the obvious mono $X' \rightarrow X$ of assemblies corresponds to a $\neg\neg$ -closed mono in $\mathcal{E}ff$; and one can show that every $\neg\neg$ -closed subobject has a representative of this form.

Another characterization of $\neg\neg$ -closed monos will be useful. Since the subobject classifier Ω of $\mathbf{Sets}$ classifies subsets of sets, the object $\nabla 2 \in \mathcal{E}ff$ classifies $\neg\neg$ -closed subobjects in $\mathcal{E}ff$; so, since $\nabla 2 \in \mathbf{Sep}$, any $\neg\neg$ -closed mono is regular in $\mathbf{Sep}$. On the other hand, one can calculate an equalizer in $\mathbf{Sep}$ by finding the equalizer on underlying sets and taking the corresponding $\neg\neg$ -closed subobject as above. To summarize,

Proposition 2.4.1 *For $X \in \mathbf{Sep}$ there is a natural correspondence between*

- (i) *$\neg\neg$ -closed subobjects of X in $\mathcal{E}ff$;*
- (ii) *regular subobjects of X in $\mathbf{Sep}$; and*
- (iii) *subsets of the underlying set of X .*

For example, if A is any set of natural numbers, we get a corresponding $\neg\neg$ -closed mono $A \rightarrow \mathbb{N}$ in $\mathcal{E}ff$.

A couple of other useful facts. Firstly, note that if ϕ and ψ are $\neg\neg$ -closed, then $\phi \vee \psi$ need not be; but there is a $\neg\neg$ -stable notion of disjunction (already used in [56] without much explanation) which sometimes serves as a replacement.

Lemma 2.4.2 *If ϕ and ψ are $\neg\neg$ -closed then $\neg\neg(\phi \vee \psi) \leftrightarrow (\neg\phi \rightarrow \psi)$.*

Proof. $\neg\neg(\phi \vee \psi) \in \Omega$ is simply the disjunction $\phi \vee_{\neg\neg} \psi$ of ϕ and ψ in the sub-Heyting algebra $\nabla 2$ of Ω . But $\nabla 2$ is a Boolean algebra so $\phi \vee_{\neg\neg} \psi$ is equivalent to $\neg\phi \rightarrow \psi$ (and also to $\neg\psi \rightarrow \phi$). ■

The following result tells us that this really is a useful notion of disjunction. We won't cite it explicitly but it explains why some of our definitions are useful ones; for example, see the discussion on partial maps and lifting in chapters 12 and 13.

Corollary 2.4.3 *Let ϕ , ψ and ξ be $\neg\neg$ -closed. If ϕ entails ξ and ψ entails ξ , then $\neg\phi \rightarrow \psi$ entails ξ .*

Proof. We know that $\phi \vee \psi$ entails ξ . But ξ is $\neg\neg$ -closed and so $\neg\neg(\phi \vee \psi)$ entails ξ . ■

Our last result will come in very handy later.

Proposition 2.4.4 *Suppose $X' \mapsto X$. Then $X' \vee \neg X' \mapsto X$ is $\neg\neg$ -dense.*

Proof. This amounts to proving that

$$((\phi \vee \neg\phi) \rightarrow \perp) \rightarrow \perp$$

is an intuitionistic tautology; which is trivial. ■

For example, $2 = 1 + 1 \mapsto \nabla 2$ is $\neg\neg$ -dense.

2.4.1 On other realizability toposes

The last cluster of results on $\neg\neg$ -closed subobjects applies to any realizability topos, as do various others that we use quite frequently: for example, the fact that maps between modest sets are always tracked by an element of the object of realizers (see later). But apart from that, we understand their internal logic rather poorly. And this logic does vary from place to place: for example, Markov's principle holds in all the toposes we'll be using, but not in the toposes arising from modified realizability.

A more surprising example is the following remark of mine which is being given substance by Edmund Robinson. In the realizability topos on (say) a term model of the λ -calculus, it is externally the case that every map from $\mathbb{N}$ to itself is tracked by an element of the object of realizers; there is also a corresponding internal formula which is valid.

However, this does not mean that Church's thesis holds in that topos. The problem is that there is no term that, given a term tracking a map $\mathbb{N} \rightarrow \mathbb{N}$,

produces a Gödel number for the corresponding recursive function; we can go effectively from codes to terms, but not backwards.

In fact, we do not even have a ‘Church’s thesis’ of the following form: “Every map from the object of realizers to itself is tracked by a term”. The problem is that there is no suitable map from the object of realizers into the (internal) object of terms.

For such a model, the syntax is not available; it vanished when we constructed the combinatory algebra by taking β -equivalence classes, so that we have, in some sense, simply an abstract λ -algebra. The effective topos is special in the following sense: the realizers are still actual codes—functions remember how they were coded up in the syntax.

This ‘explanantion’ should be taken with a grain of salt. Better to say that further research is required.

2.5 Modest sets

The internal full subcategory **Mod** in *Sep* is, of course, an internal full subcategory in $\mathcal{E}$ as well. In fact, pleasantly enough, it can be defined internally as the full subcategory of (quotients of) $\neg\neg$ -closed (internal) partial equivalence relations on **A**: that is, we could have defined

$$\mathbf{Mod}_0 = \{R \subset \mathbf{A} \times \mathbf{A} \mid R \text{ is a } \neg\neg\text{-closed PER on } \mathbf{A}\},$$

taken the generic $\neg\neg$ -closed PER $(\begin{smallmatrix} \mathbf{I} \\ \mathbf{Mod}_0 \end{smallmatrix})$, and taken the internal full subcategory (see [35]) on this **Mod**₀-indexed family of objects of $\mathcal{E}$: see [33].

So if $R, S \in \mathbf{Mod}_0$, the internal homset **Mod**(R, S) is

$$\{a \in \mathbf{A} \mid (pRp \rightarrow ap \downarrow) \wedge (pRq \rightarrow (ap)S(aq))\} / \sim$$

where

$$a \sim a' \iff \forall p. pRp \rightarrow (ap)S(ap')$$

(this is just the internal version of the description of maps $R \rightarrow S$ in **Per**).

The main thing is:

Proposition 2.5.1 *Mod is complete (relative to $\mathcal{E}$).*

This will be made more precise in the following section.

Call an object of $\mathcal{E}$ a **modest set** if it's a quotient of $\mathbf{A}$ by a $\neg\neg$ -closed partial equivalence relation; this agrees with our earlier definition of 'modest set' in *Sep*. (Over arbitrary fibres, however, we have to be more careful; see the next section.)

Here's an example, in $\mathcal{E}\mathcal{f}$, that will be useful later. If A is some nonrecursive set of natural numbers, let Z_A be the assembly with underlying set $\{\top, \perp\}$ and

$$|\top \in Z_A| = A, \quad |\perp \in Z_A| = \bar{A}.$$

This gives us a modest set in $\mathcal{E}\mathcal{f}$. Z_A can be defined in the internal logic: regarding A as a $\neg\neg$ -closed subobject of $\mathbf{N}$, let Z_A be the quotient of $\mathbf{N}$ under the $\neg\neg$ -closed (total) equivalence relation $n \in A \leftrightarrow n' \in A$.

Then it's easy to see that maps $\mathbf{N} \rightarrow Z_A$ correspond to sets of natural numbers many-to-one reducible to A ; more generally, if $X \cong \mathbf{N}/R$ is a modest set then $(Z_A)^X \cong \mathbf{N}/S$ where

$$\begin{aligned} eSe &\text{ iff } (nRn \rightarrow \{e\}(n) \downarrow) \wedge \\ &\quad (nRn' \rightarrow (\{e\}(n) \in A \leftrightarrow \{e\}(n') \in A)) \\ eSe' &\text{ iff } eSe \wedge e'Se' \wedge \forall n. nRn \rightarrow (\{e\}(n) \in A \leftrightarrow \{e'\}(n) \in A). \end{aligned}$$

So if R is a total equivalence relation then $(Z_A)^X \cong \mathbf{N}/S$ may be identified with the object of $\neg\neg$ -closed subobjects of $\mathbf{N}$ —'ordinary subsets of $\mathbf{N}$ '—that are m -reducible to A and respect the equivalence relation R .

2.6 Completeness of modest sets

[The material in this section is explained more carefully in [28] and especially [33]. Beware; there is a *lot* more to say than what's said here.]

Mod is complete in the following 'weak' sense:

Proposition 2.6.1 *Let $\mathbf{D}$ be any category in $\mathcal{E}$. Then*

$$\forall F: \mathbf{D} \rightarrow \mathbf{Mod}. \exists X \in \mathbf{Mod}. X = \varinjlim F.$$

Note that X is *not* unique—it's only unique up to isomorphism—so we can't conclude that **Mod** is complete in a 'strong' sense, namely that there is an appropriate internal functor

$$\varinjlim: \mathbf{Mod}^{\mathbf{D}} \rightarrow \mathbf{Mod}.$$

Actually, we need a somewhat stronger result; the same result as the previous one in every fibre of $\mathcal{E}$. That is, for every I -indexed family of diagrams in **Mod** (not necessarily all of the same shape) there exists an I -indexed family of limits:

Proposition 2.6.2 *Let $I \in \mathcal{E}$, let $I^*: \mathcal{E} \rightarrow \mathcal{E}/I$ be the pullback functor, and let $\mathbf{D}$ be a category in $\mathcal{E}/I$. Then in $\mathcal{E}/I$,*

$$\forall F: \mathbf{D} \rightarrow I^* \mathbf{Mod}. \exists X \in I^* \mathbf{Mod}. X = \varprojlim F.$$

This does not follow from the first result, as there is no way (in the internal language) of quantifying over all internal categories $\mathbf{D}$ in $\mathcal{E}$.

Everything is nicer when we are working entirely inside *Sep*; then **Mod** is complete in the stronger sense:

Proposition 2.6.3 *Let $J \in \text{Sep}$, and let $\mathbf{D}$ be a category in Sep/J . Then there exists an internal functor*

$$\varprojlim: (J^* \mathbf{Mod})^{\mathbf{D}} \rightarrow (J^* \mathbf{Mod})$$

taking each diagram to a limit for that diagram.

Note that all these forms of completeness are *self-dual*: **Mod** is cocomplete in the same sense in which it's complete (but it doesn't inherit colimits from $\mathcal{E}$, of course; only limits).

2.6.1 Completeness in terms of fibrations

What's the explanation of all this? There is no room to tell the whole story—see [33] and [65]—but very roughly: 2.6.2 follows from 2.6.3 because any object in $\mathcal{E}$ can be covered by an object in *Sep* (see [27],[33])—recall the interpretation of the existential quantifier in the internal language.

As for 2.6.3: Define two full subfibrations of $\mathcal{E}$ fibred over itself by

$$\begin{aligned} (\mathbf{Mod})(I) &= \text{quotients of } I^* \mathbf{N} \text{ by a } \neg\neg\text{-closed PER} \\ \{\mathbf{Mod}\}(I) &= \text{families } \left(\begin{smallmatrix} X \\ I \end{smallmatrix} \right) \text{ such that} \\ &\quad \exists R \subset_{\neg\neg} I^* \mathbf{N} \times I^* \mathbf{N}. \left(\begin{smallmatrix} X \\ I \end{smallmatrix} \right) \cong (I^* \mathbf{N})/R. \end{aligned}$$

Alternatively, a family $(\frac{X}{I})$ is in $(\mathbf{Mod})(I)$ if we have a pullback square

$$\begin{array}{ccc} X & \longrightarrow & G \\ \downarrow & & \downarrow \\ I & \longrightarrow & \mathbf{Mod}_0 \end{array}$$

and it is in $\{\mathbf{Mod}\}(I)$ if we have two pullback squares

$$\begin{array}{ccccc} X & \longleftarrow & Y & \longrightarrow & G \\ \downarrow & & \downarrow & & \downarrow \\ I & \longleftarrow & J & \longrightarrow & \mathbf{Mod}_0 \end{array}$$

where the map $J \rightarrow I$ is epi. The fibration $(\mathbf{Mod})$ is the **externalization** of the internal full subcategory $\mathbf{Mod}$ and $\{\mathbf{Mod}\}$ —which is strictly bigger—is its **stack completion** (with respect to the regular epi topology) .

The point is that $\{\mathbf{Mod}\} = \text{Orth}(\nabla 2) \cap \text{Sep}$ (in each fibre) and is hence complete as a fibration over $\mathcal{E}$ —that is, it has finite limits in each fibre and right adjoints to pullback functors which satisfy the Beck-Chevalley condition. $(\mathbf{Mod})$ does not have these properties, but it is (almost by definition) *weakly equivalent*, as a fibration, to $\{\mathbf{Mod}\}$; so it is itself complete in an appropriate (weaker) sense.

Over Sep the two fibrations are equal; they are both the same as the previous fibration $\mathbf{Mod}$ defined in section 2.2. That is, if I is any object of $\mathcal{E}$ and $(\frac{X}{I}) \in \{\mathbf{Mod}\}(I)$, and if $\alpha: J \twoheadrightarrow I$ is a map with $J \in \text{Sep}$, then $\alpha^*(\frac{X}{I}) \in (\mathbf{Mod})(J)$. And the fibrations are weakly equivalent over $\mathcal{E}$ because every object of $\mathcal{E}$ has a separated cover.

The reason this has consequences for the completeness of the internal category $\mathbf{Mod}$ is that families $(\frac{X}{I})$ in $(\mathbf{Mod})(I)$ correspond to I -indexed families $I \rightarrow \mathbf{Mod}_0$ of modest sets; so $\mathbf{Mod}$ will have (in each fibre) finite limits and indexed products ‘so far as $(\mathbf{Mod})$ has them’. In particular, given a family of diagrams in $\mathbf{Mod}$ indexed by a separated object I , there is a map with codomain I that takes each diagram to a limit cone (this comes to the same thing as 2.6.3).

Incidentally, a quick calculation shows that the fibration $(\mathbf{Mod})$ does have Π -functors right adjoint to all pullback functors, not just over Sep ; unfortunately, the authors of [33] discovered that they do not satisfy the Beck-Chevalley conditions. So this fact should be seen as a bit of a red herring.

2.7 Reflectivity

[*Note.* The sorts of results described below, like the ones just sketched above, are rather *ad hoc*; this is basically because the theory of locally small categories over a base topos has not been fully developed. In particular, there is no suitable ‘internal language’ in which we can mimic the arguments of naive category theory (whereas if we were only dealing with small categories the usual internal language would be quite adequate).

Although the difficulties are of a technical nature, they are quite genuine. However, it seems likely that these problems will be sorted out in the near future; a promising line of attack is suggested by unpublished work of Verity in the context of enriched category theory.]

Since **Mod** is small and complete, it ought to be reflective—this should follow by an appropriate version of the adjoint functor theorem.

In fact there’s a better way to see the reflection. Recall the main result of [33], that $\{\mathbf{Mod}\} = \text{Orth}(\nabla 2) \cap \text{Sep}$. Now, there’s an explicit description of the reflection into $\text{Orth}(\nabla 2)$ (see [33, proposition 2.8]); and *Sep* is also reflective, the reflection taking an object to its separated quotient. But $\text{Orth}(\nabla 2)$ is closed under quotients ([33, proposition 2.7]) so to reflect a family $(\frac{X}{I})$ into $\{\mathbf{Mod}\}(I)$ we simply reflect it into $\text{Orth}(\nabla 2)$ and take the separated quotient.

(Note that this reflection is not as nice as we’d like it to be. In particular, the fact that the separated reflection functor is not left exact is one reason why **Mod** does not have even stronger completeness properties: it seems to induce the failure of the Beck-Chevalley condition.)

A consequence is that **Mod** is **weakly reflective** in the following sense: for any object X (in any fibre),

$$\exists X' \in \mathbf{Mod}_0. \exists f: X \rightarrow X'. \forall Y \in \mathbf{Mod}_0. \forall g: X \rightarrow Y. \exists ! g': X' \rightarrow Y. g = g'f.$$

Another way of seeing this is to say that the following statement is true in the internal language: For any object X , there exists an initial object in the (small) comma category $(X \downarrow \mathbf{Mod})$.

Furthermore, over *Sep* we have a genuine cartesian reflection functor into $(\mathbf{Mod})$ (since there it coincides with $\{\mathbf{Mod}\}$).

2.7.1 Reflectivity of subcategories

Now, when is a full subcategory $\mathbf{C}$ of $\mathbf{Mod}$ reflective? Typically we will have proved that $\mathbf{C}$ is complete, and we want to apply the adjoint functor theorem.

But this is all okay! The point is that everything lives in Sep , where the necessary limits are present in a strong sense. That is: Consider the generic modest set G in $\mathcal{E}/\mathbf{Mod}_0$; form the comma category $(G \downarrow \mathbf{Mod}_0^*\mathbf{C})$; by the strong form of completeness of $\mathbf{Mod}$ (applicable because $\mathbf{Mod}_0 \in Sep$), this has an initial object given by a global section $\mathbf{Mod}_0 \rightarrow (G \downarrow \mathbf{Mod}_0^*\mathbf{C})$ over $\mathbf{Mod}_0$ which essentially gives the reflection functor $r: \mathbf{Mod} \rightarrow \mathbf{C}$ taking a modest set X to its reflection rX .

It follows that $\mathbf{C}$ will be ‘as reflective in $\mathcal{E}$ as $\mathbf{Mod}$ is’; that is, the subfibrations $(\mathbf{C})$ and $\{\mathbf{C}\}$ will have the same reflectivity properties as $(\mathbf{Mod})$ and $\{\mathbf{Mod}\}$.

Sometimes it is possible to describe the reflection more explicitly; for example, as a quotient (in the case of $\mathbf{CSig}$) or as an impredicatively defined intersection of PERs (as in chapter 13). This situation is clearly preferable; but I think it’s overoptimistic to hope that there should always be an explicit description or, if there is, to expect it to be easy to find.

Incidentally, it is also possible to deduce completeness from reflectivity in the usual way. That is: we have a diagram in $\mathbf{C}$; it has a limit in $\mathbf{Mod}$; but then this limit can actually be taken to lie in $\mathbf{C}$. The situation is somewhat delicate when we are dealing with categories $\mathbf{C}$ that are not closed under isomorphism in $\mathbf{Mod}$, as in chapter 13 (though in that case, the fact that we have a ‘canonical’ description of the reflection saves us from having to invoke an argument like this anyway).

2.8 Relative computability

We’ll refer several times to results about relative computability and Turing degrees in $\mathcal{E}ff$. Here is an account, reproduced from my paper [50]; I apologise for the future clash of notation (Σ does *not* denote the r.e. subobject classifier, as it does in the next few chapters; and ω denotes the natural numbers, not the object to be defined in section 6.1).

It was shown in [27] that the maps $\mathbf{N} \rightarrow \mathbf{N}$ are (internally or externally) just the total recursive functions. Now, a subset $A \subseteq \omega$ of natural numbers corresponds to a $\neg\neg$ -closed subobject $A \rightarrow \mathbf{N}$; let k_A be the least topology forcing A to be decidable, and let $\mathbf{N}_A$ be the sheafification of $\mathbf{N}$ with respect to this topology.

Then one would expect the maps $\mathbb{N}_A \rightarrow \mathbb{N}_A$ to be the total functions recursive in A . This is a consequence of the result proved here.

Following [27] and [31], let $\Sigma = \mathbf{P}\omega$ be the powerset of the natural numbers endowed with the following preorder: if $p, q \subseteq \omega$ then $p \vdash q$ iff there is an index e such that $\{e\}(n) \in q$ for all $n \in p$. Then Σ is a Heyting prealgebra which gives rise to a tripos on *Sets*, and the topos it represents is $\mathcal{E}ff$. Similarly, let $\Sigma_A = \mathbf{P}\omega$ with the preorder: $p \vdash_A q$ iff $\exists e. \forall n \in p. \{e\}^A(n) \in q$. This gives rise to a tripos in the same way, and hence a topos $\mathcal{E}ff_A$. The logic of Σ_A is ‘realizability with A -recursive functions’. Our main result is

Theorem 2.8.1 *$\mathcal{E}ff_A$ is a subtopos of $\mathcal{E}ff$. In fact, $\mathcal{E}ff_A \simeq Sh_{k_A}(\mathcal{E}ff)$, the topos of sheaves for the topology k_A .*

The core of the proof is the same argument that was used to prove Theorem 17.2 in [27]. First we recall some notation and facts from [27] and [31]. Suppose $p, q \subseteq \omega$, $\langle \cdot, \cdot \rangle : \omega \times \omega \rightarrow \omega$ is a standard recursive coding of pairs and $\perp$ is an index for the everywhere undefined partial recursive function. Let

$$\begin{aligned} p \wedge q &= \{\langle n, m \rangle : n \in p \text{ and } m \in q\} \\ p \rightarrow q &= \{e : \{e\}(n) \in q \text{ for all } n \in p\} \\ D_A(n) &= \begin{cases} \{\langle 0, n \rangle\} & (n \in A) \\ \{\langle 1, n \rangle\} & (n \notin A) \end{cases} \\ \psi_A(p) &= \{\langle n, e \rangle : e \in D_A(n) \rightarrow p\} \\ \psi_A^*(p) &= \bigcap \{q \subseteq \omega : p \wedge \{\perp\} \subseteq q \text{ and } \psi_A(q) \subseteq p\} \\ &= (p \wedge \{\perp\}) \cup \psi_A(p \wedge \{\perp\}) \cup \psi_A(\psi_A(p \wedge \{\perp\})) \cup \dots \end{aligned}$$

(Note that if $p \neq \phi$ and $\langle n, e \rangle \in \psi_A(p)$, then $e \neq \perp$.)

Then $\psi_A^* : \Sigma \rightarrow \Sigma$ is the ‘nucleus’ on Σ (see [61, chapter 5]) corresponding to the topology k_A , and the subsets of ω in the image of ψ_A^* (i.e. those of the form $\psi_A^*(p)$) correspond to the k_A -closed truth-values.

Let $\Sigma_{(A)} = \{\psi_A^*(p) : p \in \omega\}$ be the preorder of closed truth-values. Then it is basically enough to show that

Lemma 2.8.2 *Σ_A and $\Sigma_{(A)}$ are equivalent preorders; that is, $p \vdash_A q$ iff $\psi_A^*(p) \vdash \psi_A^*(q)$.*

Proof. Since ψ_A^* is a nucleus, $\psi_A^*(p) \vdash \psi_A^*(q)$ iff $p \vdash \psi_A^*(q)$. We may assume that $q \neq \phi$; now we can just mimic the proof in [27].

First assume $p \vdash \psi_A^*(q)$. Let e be an index such that $\{e\}(n) \in \psi_A^*(q)$ for all $n \in p$. Then we can define an A -recursive function taking elements of p to elements of q as follows: Suppose $n \in p$ and let $x = \{e\}(n)$. Then

- (i) if $x = \langle y, \perp \rangle$ with $y \in q$, output y .
- (ii) if $x = \langle m, e' \rangle$ with $e' \in D_A(m) \rightarrow \psi_A^*(q)$, then let

$$x' = \begin{cases} e'(\langle 0, m \rangle) & (m \in A) \\ e'(\langle 1, m \rangle) & (m \notin A) \end{cases}$$

Then $x' \in \psi_A^*(q)$ so we can repeat the process; it will terminate in a finite number of steps yielding an element of q as in (i). Hence $p \vdash_A q$.

Now assume $p \not\vdash_A q$. Let f be an index such that $\{f\}^A(n) \in q$ for all $n \in p$. By the second recursion theorem we can define a recursive function $e(n, y)$, where y is a code for a finite set of pieces of information (not necessarily true) of the form ' $m \in A$ ' or ' $m \notin A$ ':

$$e(n, y) = \begin{cases} \langle k, \perp \rangle & \begin{array}{l} \text{—if there is a computation } \{f\}^y(n) = k \\ \text{(using only information in } y) \end{array} \\ \langle m, g \rangle & \begin{array}{l} \text{—if the computation } \{f\}^A(n) = k \text{ asks a question} \\ \text{' } m \in A \text{'}, \text{ and this information is not in } y, \text{ and } g \text{ is an} \\ \text{index for a partial recursive function which sends} \\ \langle 0, m \rangle \mapsto e(n, y \cup \{m \in A\}) \\ \langle 1, m \rangle \mapsto e(n, y \cup \{m \notin A\}) \end{array} \end{cases}$$

Then for all n and for all y containing true information about A , $e(n, y) \in \psi_A^*(q)$. In particular, this is true if $y = \phi$ (i.e. if y contains no information).

(To see this, observe that if $\{f\}^A(n) = k$ by a computation using no information about A , then $e(n, \phi) \in q \wedge \{\perp\}$. If the computation needs to know whether $m \in A$ for one value of m , then $e(n, \phi) \in \psi_A(q \wedge \{\perp\})$; if it needs to know whether two different values of m lie in A , then $e(n, \phi) \in \psi_A(\psi_A(q \wedge \{\perp\}))$; and so on.)

But now $n \mapsto e(n, \phi)$ is a recursive function taking elements of p to elements of $\psi_A^*(q)$. Hence $p \vdash \psi_A^*(q)$. This completes the proof. ■

In the case of $\neg\neg$ -separated objects, the following proposition gives a fairly explicit description of what's going on. Its proof is left to the reader (use the same argument as above).

Proposition 2.8.3 *Let X be a $\neg\neg$ -separated object. Define another $\neg\neg$ -separated object X_A as follows: X_A has the same underlying set as X , with the membership relation*

$$|x \in X_A| = \psi_A^*(|x \in X|).$$

Then regarding X as an object of $\mathcal{E}ff_A$, X_A is its image under the equivalence $\mathcal{E}ff_A \simeq Sh_{k_A}(\mathcal{E}ff)$. Furthermore, regarding X as an object of $\mathcal{E}ff$, X_A is its k_A -sheafification, with the embedding $X \mapsto X_A$ being the identity on underlying sets (tracked by $n \mapsto \langle n, \perp \rangle$).

Corollary 2.8.4 *The space of maps $\mathbb{N}_A \rightarrow \mathbb{N}_A$ in $\mathcal{E}ff$ is the modest set $(X, \in)$ where*

$$\begin{aligned} X &= \text{set of } A\text{-recursive functions } \omega \xrightarrow{\alpha} \omega \\ |\alpha \in X| &= \psi_A^*(\text{set of indices for } \alpha) \end{aligned}$$

Proof. $\mathbb{N}_A$ is the natural numbers object in $\mathcal{E}ff_A$; so this follows from the counterpart in $\mathcal{E}ff_A$ of [27, lemma 10.1]. ■

A consequence of the theorem is the existence of many new models for polymorphism. For example, it's easy to see that we have a hierarchy of categories of modest sets, indexed by the Turing degrees: for each $A \subseteq \omega$ there is a subcategory $A\text{-Mod}$ of $\mathcal{M}od$ corresponding to the modest sets in $\mathcal{E}ff_A$. In fact these are not the only interesting internal subcategories of $\mathcal{M}od$; for we have the following connection between Turing reducibility and orthogonality (compare [33]):

Proposition 2.8.5 *$A \not\leq_T B$ if and only if $2_B \perp 2_A$.*

Proof. If $A \leq_T B$ then 2_B is the k_B -sheafification of 2_A ; in particular, there is a nonconstant map $2_A \mapsto 2_B$ induced by sheafification.

Conversely, assume that there is a nonconstant map $2_A \rightarrow 2_B$, which we may take to be the identity on underlying sets. Now, since maps $\mathbb{N} \rightarrow 2_B$ factor uniquely through $\mathbb{N}_B$, ‘relativized Church’s thesis’ in $\mathcal{E}ff_B$ tells us that they correspond to B -recursive subsets of ω ; similarly maps $\mathbb{N} \rightarrow 2_A$ correspond to A -recursive subsets of ω . In particular we have a map $\chi_A: \mathbb{N} \rightarrow 2_A$ corresponding to A . Now consider the composite

$$\mathbb{N} \xrightarrow{\chi_A} 2_A \longrightarrow 2_B.$$

This must classify a B -recursive subset of ω ; but it classifies A , so $A \leq_T B$. ■

(Note that we may replace 2_B by $\mathbb{N}_B$, or even any finite type in $\mathcal{E}ff_B$.)

Since Mod is weakly complete in $\mathcal{E}ff$, the full subcategory $Orth(2_A)$ of modest sets orthogonal to 2_A is also weakly complete (see [33]). So sitting inside the modest sets we have two ‘complementary’ hierarchies of complete internal categories in $\mathcal{E}ff$: the ‘degree A modest sets’ and the ‘ A -discrete modest sets’.

Conjecture 2.8.6 *$Orth(2_A)$ is equivalent (in some appropriate sense) to the modest sets in the realizability topos $\mathcal{E}$ obtained by performing the effective topos construction over $\mathcal{E}ff_A$.*

There are, of course, many topologies on $\mathcal{E}ff$ apart from the k_A ; in fact, no-one seems to have a clue about what the whole lattice of topologies on $\mathcal{E}ff$ might be. (A rather artificial example appears already in [61]. A more interesting one was found by van Oosten in his work on extending Lifschitz’ realizability to topos theory, in [48]; this might indicate that a characterization of the topologies on $\mathcal{E}ff$ would have great mathematical interest—but it also suggests that the problem is very difficult.) However, we do have the following result:

Proposition 2.8.7 *Let j be a topology on $\mathcal{E}ff$. If $j \geq k_A$ for all $A \subseteq \omega$, then $j \geq \neg\neg$ (hence either $j = \neg\neg$ or j is trivial: [61, chapter 5]).*

Proof. Let L be sheafification with respect to j ; by [27, proposition 16.4], it suffices to show that $L2 = \nabla 2$, i.e. that in $L2$, $|0| \cap |1| \neq \phi$.

Suppose that $|0| \cap |1| = \phi$; let $B = |0|$ and choose any $C \not\leq_T B$. Consider the composite

$$\mathbb{N} \xrightarrow{xc} 2_C \mapsto L2$$

where the second map is induced by sheafification with respect to j (it is the identity on underlying sets). If $f: \omega \rightarrow \omega$ is a total recursive function tracking this map $\mathbb{N} \rightarrow L2$, then $n \in C$ iff $f(n) \in B$; i.e. C is many-to-one reducible to B , a contradiction. ■

Chapter 3

More on polymorphic types

3.1 Preliminary remarks

Let's recap briefly. The interpretation of polymorphic types in a small complete category $\mathbf{C}$ suggested by Moggi goes as follows: If T is some type-constructor, modelled as a function $T: \mathbf{C}_0 \rightarrow \mathbf{C}_0$, then the polymorphic type

$$\Pi \alpha. \mathsf{T} \alpha$$

can be interpreted as the $\mathbf{C}_0$ -indexed product

$$\prod_{X \in \mathbf{C}_0} TX.$$

Call this the **Moggi-Hyland interpretation** of $\Pi \alpha. \mathsf{T} \alpha$ relative to $\mathbf{Mod}$; it is of course equipped with a projection map into TX , where X is any modest set.

Let's pause to see why this is actually an interpretation. The terms of a type X (with, say, a single free variable $y : Y$) are interpreted as arrows $Y \rightarrow X$ in $\mathbf{C}$. Now, if we have a uniform way of constructing a term of type TX , for any X , then in the model this corresponds to an (internally defined) family of maps $Y \rightarrow TX$, one for each X ; but then this gives rise to a uniquely determined map $Y \rightarrow \prod_{X \in \mathbf{C}_0} TX$. Conversely, given a term of type $\Pi \alpha. \mathsf{T} \alpha$ and a particular type X , we want to instantiate the term to obtain a term of type TX ; but if we have a map $Y \rightarrow \prod_{X \in \mathbf{C}_0} TX$, we can simply compose it with the projection $\prod_{X \in \mathbf{C}_0} TX \rightarrow TX$ to obtain a map $Y \rightarrow TX$. Of course, some verifications are still required.

A problem with the interpretation Note that the Moggi-Hyland interpretation does not take into account the fact that T is generally 'natural'; simply

taking the product throws away all information about naturality. An odd consequence, raised by Freyd, Phil Scott and others and also by Robinson, is that if $\mathbf{C}$ and $\mathbf{D}$ are equivalent small complete categories then the Moggi-Hyland interpretations of $\Pi\alpha.T\alpha$ relative to $\mathbf{C}$ and $\mathbf{D}$ may be quite different—simply because their objects-of-objects $\mathbf{C}_0$ and $\mathbf{D}_0$ may be completely different objects. That is,

- *The Moggi-Hyland interpretation is not stable under equivalence*

The reason is exactly that naturality data is discarded. But in fact it's not straightforward to fix this. The problem is that T often does not give rise to a functor; for example we allow $T\alpha = \alpha \rightarrow \alpha$. This is related to the notorious fact that it is very difficult to express the sense in which $X, Y \mapsto Y^X$ is natural.

Several ways of dealing with this problem have been proposed: approaches involving dinaturality (see [7]) or Freyd's theory of structors (see [19]). An odd feature of some of these results is that they are special to PER models. A more satisfying approach may emerge from unpublished work of Carboni, Kelly and Wadler.

The approach proposed in the next section is considerably simpler. It skates over some of the issues tackled head-on by the papers mentioned previously; but it does provide a very simple solution to the problem of invariance under equivalence.

3.2 The groupoid interpretation

We mentioned that type constructors $T: \mathbf{C}_0 \rightarrow \mathbf{C}_0$ were often not functors. Typically, though, we have a bifunctor

$$B: \mathbf{C}^{op} \times \mathbf{C} \rightarrow \mathbf{C}$$

(for example, $X, Y \mapsto (Y^X)^{(X^Y)}$) and on objects, T acts as follows:

$$TX = BXX.$$

This, and what follows, should look quite familiar; cf. [62].

Let $\mathbf{C}^{\cong}$ be the non-full subcategory of $\mathbf{C}$ with the same objects but only the isomorphisms as arrows. We have a diagonal functor

$$\Delta: \mathbf{C}^{\cong} \rightarrow \mathbf{C}^{op} \times \mathbf{C}$$

and hence a functor

$$T = B\Delta: \mathbf{C}^{\cong} \rightarrow \mathbf{C}$$

taking $X \mapsto BXX$.

Define the object

$$(\Pi\alpha.\tau\alpha)_{\mathbf{C}}$$

to be the limit of T . We will call this the **groupoid interpretation** of $\Pi\alpha.\tau\alpha$ in $\mathbf{C}$. Informally, one thinks of it as a product over all the isomorphism classes.

Why does this give us a sound interpretation? The main point is that if we have a family of maps $Y \rightarrow TX$ arising as the interpretation of a term of the polymorphic λ -calculus of type $\tau\alpha$ (with free type variable α), then because these maps arose in a ‘uniform’ way, they must commute with all the isomorphisms $TX \cong TX'$ arising from isomorphisms $X \cong X'$; hence they form a cone over the functor $T: \mathbf{C}^{\cong} \rightarrow \mathbf{C}$ and we get a map $Y \rightarrow (\Pi\alpha.\tau\alpha)_{\mathbf{C}}$. A full proof would involve a lengthy induction, but this fact is essentially obvious.

Stability under equivalence Now suppose that $F: \mathbf{D} \rightarrow \mathbf{C}$ is an equivalence. Then we have

$$GB(F^{op} \times F): \mathbf{D}^{op} \times \mathbf{D} \rightarrow \mathbf{D}$$

where G is the adjoint of F . Hence we get a functor

$$T': \mathbf{D}^{\cong} \rightarrow \mathbf{D}$$

and we can define, as before,

$$(\Pi\alpha.\tau\alpha)_{\mathbf{D}} = \varprojlim T'.$$

Saying that the interpretation of $\Pi\alpha.\tau\alpha$ is invariant under the equivalence F amounts to showing that

$$F(\Pi\alpha.\tau\alpha)_{\mathbf{D}} \cong (\Pi\alpha.\tau\alpha)_{\mathbf{C}}.$$

Now, F restricts to an equivalence $F: \mathbf{D}^{\cong} \rightarrow \mathbf{C}^{\cong}$, so we have a functor $TF: \mathbf{D}^{\cong} \rightarrow \mathbf{C}$; and it suffices just to prove that

$$\varprojlim TF \cong \varprojlim T.$$

This follows from

- Let $\mathbf{C}$ be a (small) complete category, $\mathbf{I}$ and $\mathbf{J}$ small categories and $F: \mathbf{I} \rightarrow \mathbf{C}$ an equivalence with adjoint $G: \mathbf{J} \rightarrow \mathbf{I}$. Then for any diagram $D: \mathbf{J} \rightarrow \mathbf{C}$, the canonical map

$$\lim_{\leftarrow} DF \rightarrow \lim_{\leftarrow} D$$

is an isomorphism.

Proof. It suffices to prove that F is initial; that is,

$$\forall J \in \mathbf{J}_0. \exists I \in \mathbf{I}_0. \text{ there is a map } FI \rightarrow J$$

and

$$\forall J \in \mathbf{J}_0. \forall I, I' \in \mathbf{I}_0. \forall f: FI \rightarrow J, f': FI' \rightarrow J.$$

there exists a path $I \rightarrow \leftarrow \cdots \rightarrow \leftarrow I'$ in $\mathbf{I}$ such that we can apply F and fill in arrows to make a commutative diagram

$$\begin{array}{ccccccc} FI & \longrightarrow & \bullet & \longleftarrow & \cdots & \longrightarrow & \bullet & \longleftarrow & FI' \\ \downarrow & & \downarrow & & & & \downarrow & & \downarrow \\ J & = & J & = & \cdots & = & J & = & J \end{array}$$

But this is immediate: given $J \in \mathbf{J}_0$ we have the counit $FGJ \rightarrow J$; and given any $I \in \mathbf{I}_0$ with $FI \rightarrow J$ we have a unique factorization

$$\begin{array}{ccc} FI & \longrightarrow & FGJ \\ & \searrow & \downarrow \\ & & J \end{array}$$

■

Alternative interpretations We could perhaps obtain different interpretations by using the same idea, but with categories other than $\mathbf{C}^{\cong}$: for example, the category of retractions in $\mathbf{C}$, or even the category $\mathbf{C}^{\leftrightarrow}$ whose objects are the same as the objects of $\mathbf{C}$ but where a morphism from X to Y consists of a pair of morphisms $X \rightarrow Y, Y \rightarrow X$ in $\mathbf{C}$.

Modelling stronger theories We have just shown how to model ML-style polymorphism. In (for example) System F , polymorphic types that look like

$$\Pi \alpha. \alpha \rightarrow (\Pi \beta. \beta \rightarrow \alpha)$$

can occur: see [23]. These can be handled inductively. The point is that although

$$X \mapsto (\Pi\beta.\beta \rightarrow X)_{\mathbf{C}}$$

no longer comes from a functor $\mathbf{C}^{op} \times \mathbf{C} \rightarrow \mathbf{C}$, it does define a functor $\mathbf{C}^{\cong} \rightarrow \mathbf{C}$; so we can do the same thing as before.

System $F\omega$ also presents no problems: here we are allowed to quantify, not just over the collection $\mathbf{C}$ of all types, but also over $\mathbf{C} \rightarrow \mathbf{C}$ and so on. But this just corresponds to taking limits in $\mathbf{C}$ indexed by the functor groupoid

$$(\mathbf{C}^{\cong})^{(\mathbf{C}^{\cong})}$$

and it's certainly the case that if $\mathbf{C}$ and $\mathbf{D}$ are equivalent then so are the corresponding endofunctor categories. So the groupoid interpretation of polymorphic types in $F\omega$ is also stable under equivalence.

The calculus of constructions and its variants present a bigger problem. How can we regard a general kind as a groupoid? I have no answer to this question at the moment.

3.2.1 Parametricity

Under more restricted circumstances one can use the description of products as intersections to prove 'polymorphic parametricity' results for the Moggi-Hyland interpretation. Such theorems are very useful when reasoning about programs whose type is polymorphic.

Roughly speaking, we say that an interpretation $|\Pi\alpha.T\alpha|$ of a polymorphic type $\Pi\alpha.T\alpha$ in a category of PERs (or modest sets) is *parametric* if it "contains only what it is forced to"; for example, an interpretation of

$$\Pi\alpha.((\alpha \rightarrow \alpha) \times \alpha) \rightarrow \alpha$$

is parametric if it contains only the Church numerals.

More generally if $T\alpha$ is of the form

$$((\alpha^{n_1} \rightarrow \alpha) \times \cdots \times (\alpha^{n_k} \rightarrow \alpha)) \rightarrow \alpha,$$

where the left-hand side looks like the signature of an algebraic data type, then $|\Pi\alpha.T\alpha|$ is parametric if it is an initial algebra for that signature. (A thing of type $\Pi\alpha.T\alpha$ is a procedure that, given a type X and an algebra structure of the

given signature on X , produces a thing of type X ; that is, the elements of $|\Pi\alpha.\tau\alpha|$ should just be the *terms* of that signature.) Results like this do in fact hold in $\mathcal{E}ff$ for the Moggi-Hyland interpretation (this amounts to saying that they hold in the classical PER model): see [34], [70].

Note that we might hope to prove parametricity results more easily for the groupoid interpretation than for the Moggi-Hyland interpretation. The reason is that the groupoid interpretation is ‘smaller’. More precisely, there is a canonical map

$$\tau: (\Pi\alpha.\tau\alpha)_{\mathbf{C}} \rightarrow \prod_{X \in \mathbf{C}_0} TX$$

from the groupoid interpretation into the Moggi-Hyland interpretation, arising from the family of projections $(\Pi\alpha.\tau\alpha)_{\mathbf{C}} \rightarrow X$; and it’s easy to see that τ is mono.

Theorem 3.2.1 (Rosolini) *The groupoid interpretation in $\mathbf{Mod}$ is parametric.*

The proof of this theorem relies on results in the sequence of papers beginning with [7]. We only give a sketch here, due to Rosolini.

The point is that we only need to show that elements of $(\Pi X.TX)_{\mathbf{C}}$ are dinatural transformations $1 \rightarrow B$. But such an element is dinatural for isomorphisms by definition; and any internal functor $\mathbf{Mod} \rightarrow \mathbf{Mod}$ (isomorphic to one which) preserves maps coded by the identity recursive function (see [71]), so it is dinatural with respect to such maps too. So all we need to prove is

Lemma 3.2.2 (Rosolini) *Any map between modest sets factors as a composite of isomorphisms and maps tracked by the identity.*

Proof. There are several proofs; one is as follows. We will talk in terms of PERs instead of modest sets (quotients of PERs).

Let R and S be $\neg\neg$ -closed PERs on $\mathbf{N}$, and suppose we have a map $f: R \rightarrow S$. Recall that the product PER $R \times S$ is defined by

$$\langle n_1, n_2 \rangle (R \times S) \langle m_1, m_2 \rangle \quad \longleftrightarrow \quad n_1 R m_1 \wedge n_2 S m_2$$

where $\langle \cdot, \cdot \rangle$ is a recursive pairing function.

The graph of f is a sub-PER $G \subset R \times S$, and clearly $R \cong G$. Enlarge G into a larger PER H by setting

$$\langle n_1, n_2 \rangle H \langle m_1, m_2 \rangle \quad \longleftrightarrow \quad f(n_1) S f(n_2).$$

(A little care is needed to make this constructive.)

H can be thought of as the image of f ; we have a map $G \rightarrow H$ tracked by the identity recursive function. But now H is isomorphic to a sub-PER K of S ; and $K \hookrightarrow S$ is tracked by the identity too. ■

In the next section we'll see how to extend the theorem to full reflective subcategories of **Mod**; this provides the first proof of parametricity for interpretations of polymorphism in a general such subcategory.

3.3 Uniformized interpretations

Let $\mathbf{C}$ be a full reflective subcategory of **Mod**, and let $F: \mathbf{Mod} \rightarrow \mathbf{C}$ be a reflection functor. If $\mathbf{C}_0$ is a constant set, we mentioned in the previous section that the Moggi-Hyland interpretation of a polymorphic type $\Pi\alpha. \top\alpha$ relative to $\mathbf{C}$ could be calculated as a $\mathbf{C}_0$ -indexed intersection of PERs. In general, however, this does not hold.

But consider the category $\bar{\mathbf{C}}$ defined as follows:

$$\bar{\mathbf{C}}_0 = \mathbf{Mod}_0$$

and if $X, Y \in \bar{\mathbf{C}}_0$,

$$\bar{\mathbf{C}}(X, Y) = \mathbf{C}(FX, FY).$$

There is a full embedding $G: \mathbf{C} \rightarrow \bar{\mathbf{C}}$ and a functor $F: \bar{\mathbf{C}} \rightarrow \mathbf{C}$ (which acts as the reflection functor F on objects and as the identity on arrows); and this defines an equivalence between $\mathbf{C}$ and $\bar{\mathbf{C}}$.

Let T be a type constructor on $\mathbf{C}$ that respects isomorphisms (as noted before, all of them generally do). Then we get a type constructor on $\bar{\mathbf{C}}$, namely GTF . The uniformized Moggi-Hyland interpretation of $\Pi\alpha. \top\alpha$ is the product

$$F\left(\prod_{X \in \bar{\mathbf{C}}_0} GTFX\right)$$

indexed by $\bar{\mathbf{C}}_0 = \mathbf{Mod}_0$. That is, we compute the Moggi-Hyland interpretation in $\bar{\mathbf{C}}$ and transport it to $\mathbf{C}$ via the equivalence.

Note that this interpretation is sound; for suppose we have an object $Y \in \mathbf{C}$. Then a map $Y \rightarrow F(\prod_{X \in \bar{\mathbf{C}}_0} GTFX)$ in $\mathbf{C}$ corresponds, under the equivalence, to a map $GY \rightarrow \prod_{X \in \bar{\mathbf{C}}_0} GTFX$ in $\bar{\mathbf{C}}$; composing with the projections gives us a

family of maps $GY \rightarrow GTFX$ and hence, using the equivalence again, a family $Y \rightarrow FGTFX \cong TFX \cong TX$ in $\mathbf{C}$ (remember that $X \cong FX$).

Conversely, suppose we have a family of maps $Y \rightarrow TX$, one for each $X \in \mathbf{C}$. Then for any $X' \in \mathbf{Mod}_0$ we have a map $Y \rightarrow TFX' \cong FGTFX'$ in $\mathbf{C}$; hence a map $GY \rightarrow GTFX'$ in $\bar{\mathbf{C}}$. But these uniquely determine a map $Y \rightarrow \prod_{X' \in \bar{\mathbf{C}}_0} GTFX'$, and thus a map $GY \rightarrow F(\prod_{X \in \bar{\mathbf{C}}_0} GTFX)$ back in $\mathbf{C}$.

The uniformized interpretation as an intersection The point here is that

$$\begin{aligned} F(\prod_{X \in \bar{\mathbf{C}}_0} GTFX) &\cong \prod_{X \in \bar{\mathbf{C}}_0} FGTFX \\ &\cong \prod_{X \in \bar{\mathbf{C}}_0} TFX \\ &\cong \prod_{X \in \mathbf{Mod}_0} TFX \end{aligned}$$

which can be calculated in $\mathbf{Mod}$; and there, since it is indexed by the constant set $\mathbf{Mod}_0$, it can be calculated as the intersection of PERs

$$\bigcap_{R \in \mathbf{Mod}_0} TFR.$$

One way of interpreting this is that it shows how $\mathbf{C}$ gives rise to a ‘submodel’ of the PER model (i.e. another model where types are PERs and polymorphic types are computed as intersections, instead of more general products or limits). It is not a submodel in the ordinary sense: rather than restricting the collection of PERs, we keep all the PERs but change the maps.

I should emphasize that this interpretation depends on the particular reflection functor F ; so if we want to use in calculations, we need to have a reasonably explicit description of F . This is not always the case (e.g. [53]).

A parametric interpretation in $\mathbf{C}$ We can combine the two interpretations by defining the **uniformized groupoid interpretation** computed by applying F to a certain limit in $\bar{\mathbf{C}}$ indexed by $\bar{\mathbf{C}}^{\text{eq}}$. As in the end of the last section, one can show that this interpretation is *parametric*; the main point is that the object-of-objects of $\bar{\mathbf{C}}$ is $\mathbf{Mod}_0$, a constant set, so the results of [71] can be used. Details will appear in a forthcoming paper by Robinson and Rosolini.

The closure of a category Let me mention a red herring. Given a full subcategory $\mathbf{C}$ of $\mathbf{Mod}$ with $\mathbf{C}_0 \subset \mathbf{Mod}_0$, one can consider the full subcategory $\mathbf{C}_{\neg\neg}$ of $\mathbf{Mod}$ whose object-of-objects is $\neg\neg\mathbf{C}_0$ (the double-negation closure of $\mathbf{C}_0$ in $\mathbf{Mod}_0$);

intuitively, an object of $\mathbf{C}_{\neg\neg}$ is a PER in $\mathbf{C}$, but without the intensional information telling us that the PER is actually in $\mathbf{C}$. Then $\mathbf{C}$ is still a full reflective subcategory of $\mathbf{C}_{\neg\neg}$; but there seems to be no reason why the two categories should be equivalent, without extra assumptions.

For example, it suffices that the reflection $F: \mathbf{Mod} \rightarrow \mathbf{C}$ satisfies $FX = X$ for $X \in \mathbf{C}$: equality, not isomorphism. But this is not usually the case unless $\mathbf{C}_0$ was a constant set in the first place, and R was calculated as an intersection: for example, this was the situation in [56].

Chapter 4

Fibrations and subtypes

4.1 Introduction

4.1.1 Introductory remarks

At the centre of object-oriented programming, and related approaches to programming, are the notions of subtyping and inheritance. These have proved to be very powerful tools for structuring programs, and they appear—in one form or another—in a wide variety of modern programming languages.

One way of studying these notions formally is to use the framework of typed λ -calculus. That is, we start with a formal system (say, a version of the polymorphic λ -calculus) and extend it by adding a notion of type inclusion, together with suitable rules; we obtain a larger system. We can then use the methods of mathematical logic to study the properties of the system.

Category theory provides a useful general framework for talking about the semantics of typed λ -calculi, and the notion of model. For example, models of the simply typed λ -calculus correspond to cartesian closed categories; and models of various polymorphic λ -calculi correspond to various sorts of fibrations (or indexed categories, if you prefer). That is, to construct a model we need only construct an appropriate sort of (possibly fibred) category.

In fact, the categorical structures that arise can be studied without reference to any formal systems. For example, we can study cartesian closed categories quite independently of the simply typed λ -calculus; the theory can be viewed as an alternative formulation of the same notions, less suitable as a programming language but more amenable to mathematical (rather than logical or syntactic)

analysis.

This chapter is a tentative attempt to find a corresponding categorical framework for the polymorphic λ -calculus with subtyping and bounded quantification. That is, the emphasis is on defining appropriate categorical structure; our definitions are motivated by certain rules appearing in existing formal systems such as Quest and Ponder, but the formalism of typed λ -calculus plays a strictly secondary role; in fact, no complete formal systems are presented.

For a category theorist, as opposed to a logician, this is the more natural point of view. We'll also see that it's a useful one: for example, it leads to the discovery that various categories of 'synthetic domains' can be used to model subtyping. We also describe a couple of simple consequences of our axioms that have some type-theoretic meaning; they are obtained by categorical, rather than logical, means.

Throughout this chapter we rely heavily on the language of fibrations (see [10] and [33]). In contrast with the rest of the thesis the internal language of a topos is used nowhere; nonetheless, it was a useful tool in obtaining the results of the final section, where we are dealing with particular categories that live inside realizability toposes (see [28], [33] for an explanation of the connection). I am not sure of a good way to talk about subtyping in the internal language. One reason is the following:

A funny feature There is an aspect of these proposals which will seem very odd to a category theorist, but which seems to be unavoidable at the moment. Namely: our notion of 'inclusion' is not stable under isomorphism.

This is a basic feature of the PER model: there types $X, Y, \dots$ are modelled by partial equivalence relations $R, S, \dots$, and we say that $X \leq Y$ (' X is a subtype of Y ') if $R \subseteq S$ —a relation that is clearly not preserved when we compose with isomorphisms. And this does reflect a pragmatic 'fact': subsumption or coercion should involve a trivial amount of computation, preferably none at all, whereas isomorphism can be arbitrarily complex functions.

However, this does make the category theory look a bit unnatural. We are forced to talk about equality (rather than just isomorphism) of objects; we have to use split (rather than cloven) fibrations, so that reindexing functors commute on the nose... This is all possible, of course, but it requires some extra care. Furthermore, it prevents us from giving a version that makes sense in the internal logic of a topos.

Actually, the definitions that follow can be ‘loosened’ to give a notion of subtyping that is stable under composition with isomorphisms. In fact, they were phrased with this possibility in mind. I won’t do this, though, as I don’t know how to. More speculatively, one might hope to prove a coherence theorem stating that any ‘loosened’ structure was equivalent to a ‘strict’ one.

Unanswered questions There are many; let me just mention a couple problems:

- Integrating subtyping with recursive types (as in Quest): see [5].
- Extending subtyping to the full calculus of constructions.
- Using the framework to find a better encoding of records (this is work in progress).
- Seeing how other models, such as strict maps or ideal or retraction models, fit in.

All are the subject of current research and will not be discussed further here. They are problematic to say the least.

Note. The notation in this chapter is a little inconsistent with the rest of the thesis: **P**, not **Mod**, denotes the category of modest sets. In fact, this usage follows [33].

4.1.2 Fibrations and polymorphic λ -calculus

We’ll be concerned with systems like $F\omega$ (plus subtypes and bounded quantification) or a weakened version of Quest: cf. [13]. But I won’t set down any formal systems here; instead, I’ll just give an informal description. So some familiarity with typed λ -calculus is already assumed.

Here are the rough features of our system.

- We have *kinds* $I, J, \dots$ which are closed under the formation of product kinds and arrow kinds: $I \times J, I \rightarrow J$.
- We have *types* $S, T, \dots$ which are closed under the formation of product types and arrow types: $S \times T, S \rightarrow T$.

- We can have families of types indexed by kinds: $T(X) [X \in \mathbf{I}]$. In particular, there is a kind $\mathbf{T}$ of types with a ‘generic type’ $G(X) [X \in \mathbf{T}]$ indexed by $\mathbf{T}$.
- We can form products of families of types indexed by kinds; for example, given the type $T(X) [X \in \mathbf{I}]$ we can form the polymorphic type $\Pi X \in \mathbf{I}.T(X)$.
- We have appropriate *terms* $t : T$ of each type.

Note that we take products as primitive, rather than encoding $S \times T$ as

$$\Pi X \in \mathbf{T}.(S \rightarrow T \rightarrow X) \rightarrow X;$$

from a category-theoretic point of view, our approach seems more natural (and saves us from having to prove that the encoding actually has the property of being a categorical product).

One can use fibrations to model systems like this; see, most recently, the work of Bart Jacobs. There is no room to outline the theory of fibrations here, but see [10] for a long discussion or [33] for a fairly brief summary.

The sort of structure we need is as follows:

1. A cartesian closed category $\mathcal{K}$ of *kinds*.
2. A fibration $\mathcal{T}^{\mathbf{I}}_{\mathcal{K}}$ over $\mathcal{K}$ of *types*, or rather families of types indexed by kinds. This fibration must have the following additional structure:

- Each fibre $\mathcal{T}^{\mathbf{I}}$ over each kind $\mathbf{I} \in \mathcal{K}$ is cartesian closed.
- The fibration is cloven, so that given any $f:\mathbf{I} \rightarrow \mathbf{J}$ in $\mathcal{K}$ we have a ‘substitution’ or ‘reindexing’ functor $f^*:\mathcal{T}^{\mathbf{J}} \rightarrow \mathcal{T}^{\mathbf{I}}$; furthermore, each f^* should preserve products and exponentials.
- The fibration is complete: that is, we are given a right adjoint Π_f for each substitution functor f^* , satisfying the Beck-Chevalley condition.
- There is a kind $\mathbf{T}$ and a ‘generic type’ $G \in \mathcal{T}^{\mathbf{T}}$ such that any type $X \in \mathcal{T}^{\mathbf{I}}$ is (isomorphic to) f^*G for a unique arrow $f:\mathbf{I} \rightarrow \mathbf{T}$.

For example, we could take $\mathcal{K} = \mathbf{Sets}$, and let $\mathcal{T}^{\mathbf{I}}$ be the category of $\mathbf{I}$ -indexed families of PERs and uniform families of maps. This gives the usual PER model for $F\omega$.

Later on we’ll want our fibration to have stronger properties, as well as extra structure.

Σ -types To handle abstract data types, and for other reasons, it's useful to have types of the form $\Sigma X.T(X)$; these corresponds to *left* adjoints to substitution functors (and hence, like binary products, should be taken as primitive rather than defined in terms of Π and $\rightarrow$). I believe that they can be fitted into the framework proposed in this chapter; but, for simplicity, we won't discuss them further.

4.2 Modelling subtypes using fibrations

4.2.1 Rules for subtypes and bounded quantification

To motivate the categorical definitions that will appear soon, we need to consider the sort of rules that we'd like a notion of subtyping to satisfy. The following rules (not spelled out precisely!) are based on the ones in Ponder and Quest, and many other systems.

A subtyping judgement that looks roughly like

$$\Gamma \vdash S \leq T$$

where S, T are types and Γ is a list of subtyping constraints of the form

$$X \leq U \quad \text{or} \quad U \leq X$$

where U is a type and X is a type variable. Of course S, T and the types U can also be families of types varying over a fixed kind $\mathbf{l}$.

In the rules that follow we usually leave out Γ . We also leave out variable restrictions, and often any mention of types being explicitly indexed over kinds—for example, we will write $\Pi X.T(X)$ instead of $\Pi X \in \mathbf{l}.T(X)$ when the rule in question can be understood without knowing anything about the particular kind $\mathbf{l}$.

The first rule is the *subsumption* rule:

$$\frac{t : S \quad S \leq T}{t : T}$$

One could replace this rule by some *coercion* rules (see [13]); this approach would be easier to model, but we do not consider it here.

Type inclusion should be reflexive and transitive; we would also like to have the *contra-co* subtyping rule

$$\frac{S \leq T \quad S' \leq T'}{T \rightarrow S' \leq S \rightarrow T'}$$

as well as several rules relating subtypes to polymorphic types: for example,

$$\frac{}{\Pi X. T(X) \leq T[X := S]}$$

$$\frac{}{\Pi X. T \rightarrow S(X) \leq T \rightarrow \Pi X. S(X)}$$

$$\frac{T \leq S(X)}{T \leq \Pi X. S(X)}$$

To encode records (see [13]) we need **Top** and rules for products. **Top** is a type such that

$$T \leq \mathbf{Top}$$

for any type T . And for products, we simply need

$$\frac{S \leq S' \quad T \leq T'}{S \times T \leq S' \times T'}$$

as well as

$$\frac{U \leq S \times T \quad t : U}{\text{fst}(t) : S}$$

and similarly for **snd**.

Having bounded quantification means being able to form polymorphic types like $\Pi X \leq T. S(X)$. It clearly suffices to have, for each type T , a *power kind* $\downarrow T$, ‘the kind of subtypes of T ’; we could then just form Π -types indexed by $\downarrow T$; that is, types having the form $\Pi X \in \downarrow T. S(X)$.

[One might also like to form the ‘kind of supertypes’ $\uparrow T$, and hence polymorphic types of the form $\Pi T \leq X. S(X)$; such things can be modelled, but they don’t seem to be very useful.]

All this has to interact properly with the fact that there’s a generic type G (over $\mathbf{T}$) with associated power kind $\downarrow G$.

Rules for subkinds In Quest there is also a notion of subkind, which we might write $\mathbf{I} \subseteq \mathbf{J}$. We won’t discuss it in any detail here, but simply indicate briefly how it fits into the framework. It raises some issues that I haven’t thought through yet.

The subkind relation should be reflexive and transitive; it should probably behave well with respect to products of kinds, but it need not satisfy the contra-co rule. It is related to subtyping via the following rules:

$$\frac{S \leq T}{\downarrow S \subseteq \downarrow T}$$

$$\frac{I \subseteq J}{\Pi X \in J.S(X) \leq \Pi X \in I.S(X)}$$

There is a kind subsumption rule in Quest which is not considered here. In fact, we will soon see that the notion of subkind may not be necessary in our framework.

4.2.2 Characterizing inclusions

From the point of view of fibrations, what could a model of such a system look like? As I mentioned earlier, we need to be able to model notions of type inclusion that are not stable under isomorphism, and this requires some care.

So as before, we'd like a fibration $\mathcal{T}$ of the sort described earlier. But we'll also need to impose the following conditions:

split The fibration is split: so if $f: I \rightarrow J$ and $g: J \rightarrow K$ then $(gf)^* = f^*g^*$ (for a cloven fibration, we'd only have $(gf)^* \cong f^*g^*$).

Π -functors compose Π -functors should also compose 'on the nose': that is, $\Pi_{gf} = \Pi_g \Pi_f$.

[Bob Walters pointed out that instead of using a split fibration, we might as well just talk about an ordinary functor $\mathcal{K} \rightarrow \mathcal{Cat}$. We prefer to phrase everything in terms of fibrations so that we can generalise to possible notions of inclusion that may be stable under isomorphism, and because some people prefer talking about fibrations rather than indexed categories anyway.]

Subtypes are modelled as follows: For each fibre T^I over I , we are given a class of maps

$$S \hookrightarrow T$$

in T^I , called **inclusions**; such a map witnesses the fact that S is a subtype of T . These classes of maps should satisfy the following conditions:

uniqueness For any objects S, T , there is at most one inclusion $S \hookrightarrow T$ (this seems necessary to get a coherent interpretation of subsumption).

stability Substitution functors f^* preserve inclusions.

identities Every identity map is an inclusion (though an isomorphism need not be an inclusion).

composition A composite of inclusions is an inclusion.

The last three conditions say that the inclusions form a (non-full) subfibration of the fibration of types over kinds; the first, that they form a fibred preorder over $\mathcal{K}$.

Now we need to say how inclusions must behave with respect to arrow types, binary products and **Top**.

contra-co If $S \hookrightarrow T$ and $S' \hookrightarrow T'$ are inclusions, then so is the induced map $S'^T \rightarrow T'^S$.

products If $S \hookrightarrow S'$ and $T \hookrightarrow T'$ are inclusions, then so is the induced map $S \times T \rightarrow S' \times T'$.

top There is a distinguished terminal object **Top** in each fibre, such that $f^*\mathbf{Top} = \mathbf{Top}$ for any $f: \mathbf{I} \rightarrow \mathbf{J}$ in $\mathcal{K}$, and such that for any object T in any fibre, the unique arrow $T \rightarrow \mathbf{Top}$ is an inclusion.

And inclusions must behave properly with respect to products. Let $f: \mathbf{I} \rightarrow \mathbf{J}$ be any arrow in $\mathcal{K}$, and let $S \in T^{\mathbf{I}}$, $T \in T^{\mathbf{J}}$ be any two objects in the fibres over $\mathbf{I}$ and $\mathbf{J}$; then:

counits The component $\epsilon_S: f^*\Pi_f S \rightarrow S$ of the counit of $f^* \dashv \Pi_f$ at S is an inclusion.

mates If $f^*T \hookrightarrow S$ is an inclusion, then so is its mate $T \rightarrow \Pi_f S$.

strengths Recall that f^* preserves products, and consider the map

$$f^*(T \times \Pi_f(S^{f^*T})) \cong f^*T \times f^*\Pi_f(S^{f^*T}) \rightarrow f^*T \times S^{f^*T} \rightarrow S$$

where the middle map is $f^*T \times \epsilon_{S^{f^*T}}$ and the rightmost map is evaluation; let its mate be $T \times \Pi_f(S^{f^*T}) \rightarrow \Pi_f S$; then the corresponding map $\Pi_f(S^{f^*T}) \rightarrow (\Pi_f S)^T$ is an inclusion.

Of course, all of these conditions are just categorical formulations of the subtyping rules sketched above.

There is a nice—if unsurprising—feature of our axioms. Namely, we never need to assume the existence of new maps between types that need not otherwise be there; our conditions only say that certain *existing* maps should be inclusions.

4.2.3 Incorporating bounded quantification

For powerkinds we only need a generic subtype for the generic type $G \in T^{\mathbf{T}}$. First we need pullbacks in the category of kinds (which will allow us to obtain arbitrary powerkinds from the powerkind of the generic type G). We also want G to be a generic type in the ‘strict’ sense.

pullbacks $\mathcal{K}$ has pullbacks.

strict generic For any $\mathbf{I} \in \mathcal{K}$ and any $T \in T^{\mathbf{I}}$ there is a unique map $\eta: \mathbf{I} \rightarrow \mathbf{T}$ with $T = \eta^*G$.

Now we must have a ‘generic powerkind’. That is, we must have a kind $\downarrow G \in \mathcal{K}$, a map $\alpha: \downarrow G \rightarrow \mathbf{T}$, an object $G' \in T^{\downarrow G}$ and an inclusion $G' \hookrightarrow \alpha^*G$ satisfying the following universal property:

powerkinds Let $T \in T^{\mathbf{I}}$, and define the kind $\downarrow T$ by the pullback square

$$\begin{array}{ccc} \downarrow T & \xrightarrow{\epsilon} & \downarrow G \\ \epsilon \downarrow & & \downarrow \alpha \\ \mathbf{I} & \xrightarrow{\eta} & \mathbf{T} \end{array}$$

so that we have an inclusion $T' \hookrightarrow \xi^* \alpha^* G = \epsilon^* T$ where $T' = \xi^* G'$; then if $f: \mathbf{J} \rightarrow \mathbf{I}$ and $S \hookrightarrow f^* T$ is any inclusion in $T^{\mathbf{J}}$ then there is a unique map $\kappa: \mathbf{J} \rightarrow \downarrow T$ such that $\epsilon \kappa = f$, $S = \kappa^* T'$ and

$$S = \kappa^* T' \hookrightarrow \kappa^* \epsilon^* T = f^* T$$

is the same map as $S \hookrightarrow f^* T$.

One can define generic supertypes in a completely symmetrical way; however, we don’t bother to do this.

Power kinds without a generic type In the absence of G , we can still have powerkinds: instead of defining the kind $\downarrow T$ via a pullback as above, simply say that for each type $T \in \mathcal{T}^{\mathbf{I}}$ we must have a kind $\downarrow T$, a map $\epsilon: \downarrow T \rightarrow \mathbf{I}$ with an inclusion $T' \hookrightarrow \epsilon^*T$ in the fibre $\mathcal{T}^{T'}$ —all satisfying an appropriate universal property (the one stated above, in fact).

The details are left to the reader. Note that the definition looks a bit like the definition of locally small fibration. Note also that our conditions do require the existence of ‘new’ arrows in $\mathcal{K}$; as for **top**, these new arrows crop up because we have added new objects (this time to $\mathcal{K}$, not $\mathcal{T}$).

4.2.4 Incorporating subkinds

Building a notion of subkind into our structure is very similar. That is, we must single out a class of ‘inclusion maps’ in $\mathcal{K}$ satisfying certain conditions. Writing $\mathbf{I} \sqsubseteq \mathbf{J}$ for an inclusion of kinds, we can state these as follows:

subkind-uniqueness There is at most one kind inclusion between any two kinds.

subkind-identities Identity maps are kind inclusions.

subkind-composition The composite of two kind inclusions is a kind inclusion.

subkind-powerkinds If $S \hookrightarrow T$ is a type inclusion, there is a kind inclusion $\downarrow S \sqsubseteq \downarrow T$ in $\mathcal{K}$.

subkind-restrictions If $m: \mathbf{I} \sqsubseteq \mathbf{J}$ is a kind inclusion, $f: \mathbf{J} \rightarrow \mathbf{K}$ is any map and $T \in \mathcal{T}^{\mathbf{J}}$, there is a type inclusion $\Pi_f T \hookrightarrow \Pi_{fm} m^*T$.

Proposition 4.2.1 *The subkind-restrictions condition is redundant; it holds for any map $m: \mathbf{I} \rightarrow \mathbf{J}$ between kinds.*

Proof. We will define a natural map $\Pi_f T \rightarrow \Pi_{fm} m^*T$ and show that it’s a type inclusion.

Firstly, by the **counts** condition we have a type inclusion $f^* \Pi_f T \hookrightarrow T$; by the **stability** condition we can apply m^* to obtain a type inclusion $(fm)^* \Pi_f T = m^* f^* \Pi_f T \hookrightarrow m^* T$; and by the **mates** condition, the corresponding map $\Pi_f T \rightarrow \Pi_{fm} m^*T$ must be a type inclusion. ■

The **subkind-powerkinds** condition is sort of redundant too; a candidate for the kind inclusion already exists:

Proposition 4.2.2 *If $S \hookrightarrow T$ is a type inclusion, there is a canonical map of kinds $\downarrow S \rightarrow \downarrow T$.*

Proof. Suppose $S \hookrightarrow T$ is a type inclusion in $\mathcal{T}^1$, and let $\phi: \downarrow S \rightarrow \downarrow T$ be the arrow guaranteed to exist by **powerkinds**; then by **stability** we have an inclusion $\phi^*S \hookrightarrow \phi^*T$ in $\mathcal{T}^{1S}$. So by **powerkinds** again, there is a unique arrow $\kappa: \downarrow S \rightarrow \downarrow T$ such that $\kappa^*(T' \hookrightarrow \epsilon^*T) = \phi^*S \hookrightarrow \phi^*T$. ■

So our categorical definitions can actually be used to prove (small) theorems. In this case, we have shown that the notion of kind inclusion is not really necessary; if we are prepared to forgo the **kind-uniqueness** condition (and hence kind subsumption), we may as well consider arbitrary maps of kinds to be kind ‘inclusions’—the rules we want will still be valid.

We might like to do the same for type inclusions. There are two problems:

- This idea does not seem computationally realistic (type inclusions ought to be computationally trivial; but then, what about kind inclusions?); and
- If we allow all arrows between types to be type inclusions, powerkinds may not exist—indeed, I don’t know of any models in which they do, at least for which $\mathcal{K}$ is *Sets*.

4.3 Examples

4.3.1 PER models

We will see how the classical PER models for subtyping fit into the framework described above. Let A be a partial combinatory algebra—for example, $\mathbf{N}$ (indices for partial recursive functions) or some model of the untyped λ -calculus. We will model types as partial equivalence relations (or **PERs**) on A . All this is essentially well-known; it’s only the category theory that’s new.

For PERs R, S on A there are, at first sight, at least two possible notions of inclusion:

- The usual one:

$$aRb \longrightarrow aSb;$$

that is, $R \subseteq S$.

- A stronger one: $R \subseteq S$ and

$$aSb \wedge aRa \wedge bRb \longrightarrow aRb$$

as well.

(There may be others, for special $\mathbf{A}$.)

Recall that PERs correspond, roughly speaking, to modest sets in the realizability topos on $\mathbf{A}$. Then the second notion corresponds to the notion of $\neg\neg$ -closed monomorphism between modest sets; that is, every $\neg\neg$ -closed subobject of a modest set has a unique representative of that form. So it has a good categorical characterization. Unfortunately, it does not seem to be a very useful notion: for example, if we were to adopt it, the contra-co subtyping rule would fail.

So by inclusion of PERs we mean the first sort of inclusion. Speaking categorically, this is not such a natural notion; for example, Rosolini showed that *any* map between two PERs factors as a composite of isomorphisms and inclusions (see [57]). However, it turns out to be pretty useful—it gives a simple model of subtyping that’s pretty easy to work with. In fact, it’s the usual notion: see [13].

First we have to define a fibration $\frac{\mathcal{P}}{\mathcal{K}}$.

- $\mathcal{K}$ is *Sets*, the ordinary category of sets.
- If $\mathbf{I}$ is a set, the fibre $\mathcal{P}^{\mathbf{I}}$ of $\mathcal{T}$ over $\mathbf{I}$ is the category of $\mathbf{I}$ -indexed families of PERs on $\mathbf{A}$ and uniform families of maps. Products and exponentials are calculated in the standard way for PERs.
- The kind of type $\mathbf{T}$ is the set $\mathbf{P}$ of all PERs on $\mathbf{A}$.
- The generic type $G \in \mathcal{P}^{\mathbf{P}}$ is the family $\{R\}_{R \in \mathbf{P}}$ of all PERs on $\mathbf{A}$.

This has the ‘strict’ structure required. (Recall that Π -functors are calculated as ordinary intersections of PERs: see [45] and [27]).

Let $T' = \{R_i\}_{i \in \mathbf{I}}$, $T = \{S_i\}_{i \in \mathbf{I}} \in \mathcal{P}^{\mathbf{I}}$ be two $\mathbf{I}$ -indexed families of PERs; then there is an inclusion $T' \hookrightarrow T$ exactly when $R_i \subseteq S_i$ for all $i \in \mathbf{I}$; that is, an inclusion is a uniform family of maps $R_i \rightarrow S_i$ tracked by the identity.

To handle powerkinds we have:

- $\downarrow G = \{\langle S, R \rangle : S, R \text{ are PERs and } S \subseteq R\}$
- $\alpha : \downarrow G \rightarrow \mathbf{P}$ is the projection $\langle S, R \rangle \mapsto R$

$G' \in \mathcal{P}^{kG}$ and $G' \hookrightarrow \alpha^* G$ are obvious.

All the conditions can be verified by direct calculation, if products and exponentials are defined as PERs in the usual way and Π -functors are defined using intersections. The details are omitted.

Observe that if $\{R_i\} \in \mathcal{P}^1$ then the powerkind $\downarrow \{R_i\}$ is the set

$$\{\langle i, S, R_i \rangle : S \subseteq R_i\}.$$

This is what we ought to expect.

Note the problem in extending all this to the calculus of constructions: Now we must index, not just over *Sets*, but over $\mathbf{A}$ -valued assemblies (e.g. ω -sets); and Π -functors are no longer calculated simply as intersections.

Interpreting subkinds We just note here that kind inclusion $\mathbf{I} \subseteq \mathbf{J}$ can be interpreted as ordinary subset inclusion. The only problem is the **subkind-powerkinds** condition; we may need to redefine powerkinds to get an actual subset inclusion.

4.3.2 Models arising from PER models

Sometimes we're interested, not in all PERs on $\mathbf{A}$, but only in certain special ones: for example, ones that can be regarded as '(pre)domains': see, e.g., [53]. That is, if (by abuse of language) we let $\mathbf{P}$ denote the category of modest sets in the realizability topos on $\mathbf{A}$, then we're interested in a full subcategory

$$\mathbf{C} \hookrightarrow \mathbf{P}$$

of $\mathbf{P}$. The object-of-objects need not be a 'set' ($\neg\neg$ -sheaf or 'uniformizable' object: see [27]); it could be any subobject of the object of modest sets. We'll consider the situation when this category is *reflective*: cf. [55].

Then we have a map

$$r: \mathbf{P} \rightarrow \mathbf{P}$$

that is the object part of the internal functor $\mathbf{P} \rightarrow \mathbf{C} \rightarrow \mathbf{P}$, where the first arrow is the internal reflection functor and the second arrow is the inclusion.

Proposition 4.3.1 *Exponentials and indexed products in $\mathbf{C}$ agree with the ordinary ones for PERs (modest sets); we may compute the indexed products or exponentials of objects of $\mathbf{C}$ in the category of modest sets, so that the resulting objects again lie in $\mathbf{C}$ and are products/exponentials in that category.*

Proof. This is a well-known fact about full reflective subcategories. ■

Then we can define a new model using a construction appearing in [57]: that is, we define a fibration $\downarrow_{\mathcal{K}}^{\mathcal{C}}$ as follows.

- $\mathcal{K}$ is *Sets* again.
- $\mathcal{C}^{\mathbf{I}}$ is the category of $\mathbf{I}$ -indexed families of PERs, where maps $\{R_i\} \rightarrow \{S_i\}$ are uniform families of maps $rR_i \rightarrow rS_i$.
- $\mathbf{T} = \mathbf{P}$ as before.

Note that Π -functors, binary products and exponentials in each fibre, can be calculated very simply; for example,

$$\{S_i\}^{\{R_i\}} = \{rS_i^{rR_i}\}$$

and

$$\{R_i\} \times \{S_i\} = \{rR_i \times rS_i\}.$$

We'll illustrate indexed products by calculating the Π -functor along a map $\mathbf{I} \rightarrow 1$: Let $\{R_i\} \in \mathcal{C}^{\mathbf{I}}$ and let S be a PER. Then uniform families of maps of PERs $rS \rightarrow rR_i$ correspond to maps $rS \rightarrow \Pi_i rR_i$ of PERs, where Π_i is the $\mathbf{I}$ -indexed product of ordinary PERs: we could define $\Pi_i\{rR_i\} = \bigcap_i rR_i$.

So maps $\{S\} \rightarrow \{R_i\}$ in $\mathcal{C}^{\mathbf{I}}$, i.e. uniform families of maps of PERs $rS \rightarrow rR_i$, correspond to uniform families of maps of PERs $rS \rightarrow \Pi_i rR_i \cong r\Pi_i rR_i$; that is, to maps $S \rightarrow \Pi_i rR_i = \bigcap_i rR_i$ in $\mathcal{C}^1$. Therefore, in the fibration $\downarrow_{\mathcal{K}}^{\mathcal{C}}$, we can define the Π -functor for $\mathbf{I} \rightarrow 1$ as $\Pi_i\{R_i\} = \bigcap_i rR_i$.

[Note that we *cannot* put $\Pi_i\{R_i\} = \bigcap_i R_i$.]

The **inclusions** are the families of maps tracked by the identity; that is, we have an inclusion $\{R_i\} \hookrightarrow \{S_i\}$ if

$$rR_i \subseteq rS_i$$

for all $i \in \mathbf{I}$. And if we put

$$\downarrow G = \{\langle S, R \rangle : rS \subseteq rR\}$$

and so on, it all carries through. The main observations are

- ' $rS \subseteq rR$ ' is a $\neg\neg$ -closed predicate, so $\downarrow G$ can be regarded as a set and not just an assembly (i.e. ω -set); and

- products and exponentials may be calculated in basically the same way as for PERs, and so satisfy the required properties for the same reasons as before.

(The first remark was not used explicitly; it simply explains why the constructions work, and why things don't work if we allow arbitrary arrows to be type inclusions—because then the resulting ‘powerkinds’ are no longer ordinary sets, i.e. $\neg\neg$ -sheaves.)

In [71], Rosolini showed that if the object-of-objects of $\mathbf{C}$ was a $\neg\neg$ -sheaf then $\mathbf{C}$ gives rise to a model of subtyping; so we have basically generalized Rosolini’s result to arbitrary reflective $\mathbf{C}$.

There should be other models. For example, we should be able to allow stability under isomorphism (in a certain sense) so that categories of synthetic domains form a model where the inclusions are *strict* maps—making a connection with the work of Rosolini. In this case we could not expect powerkinds unless we took *Sep*, rather than *Sets*, as the category of kinds.

4.3.3 Subtypes and constructions on types

If $\mathbf{C}$ can be regarded as a category of predomains, it will probably support various other constructions: for example,

$$\begin{aligned} T &\mapsto T_{\perp} && \text{lifting (computable partial map classifier)} \\ T &\mapsto PT && \text{(convex, or some other) powerdomain} \end{aligned}$$

We might then like to have rules like

$$\frac{S \leq T}{S_{\perp} \leq T_{\perp}} \qquad \frac{S \leq T}{PS \leq PT}$$

That is, we’d like the functors $(-)\perp$ and $P(-)$ to preserve inclusions.

Now $T_{\perp}$ and PT are characterized by universal properties, i.e. only (at first sight) up to isomorphism; so why should they preserve inclusions, which may not be stable under isomorphism? This is a general problem; luckily, in the case of PERs, we know how to handle it.

The solution is in [71]: Rosolini showed using the uniformity principle that if $\mathbf{C}$ is a full subcategory of $\mathbf{P}$ whose object-of-objects is a $\neg\neg$ -sheaf, then any internal functor $\mathbf{C} \rightarrow \mathbf{C}$ may be assumed to preserve inclusions. One can deduce

from this that fibred functors $\overset{\mathcal{C}}{\underset{\mathcal{K}}{\downarrow}} \rightarrow \overset{\mathcal{C}}{\underset{\mathcal{K}}{\downarrow}}$ arising from internal functors must preserve inclusions.

The fibration just described was essentially the externalization of the category $\bar{\mathbf{C}}$, equivalent to $\mathbf{C}$, described in [57]; its object-of-objects is simply $\mathbf{P}$, so Rosolini's result applies here too. That is, for any full reflective subcategory $\mathbf{C}$ of $\mathbf{P}$, 'internally definable' fibred endofunctors $\overset{\mathcal{C}}{\underset{\mathcal{K}}{\downarrow}} \rightarrow \overset{\mathcal{C}}{\underset{\mathcal{K}}{\downarrow}}$ preserve inclusions.

Chapter 5

Σ -spaces as predomains

5.1 The dominance Σ

Enough distractions; we should start trying to find some domains in $\mathcal{E}ff$. This section explains some essential preliminaries. The material is not original; it's based on ideas due to Mulry, Scott and Rosolini. See [68].

Recall that maps $\mathbf{N} \rightarrow 2$ in $\mathcal{E}ff$ correspond to recursive subsets. Is there a subobject of Ω that will classify the *recursively enumerable* subobjects of $\mathbf{N}$? There is; and it's the linchpin of everything that follows.

Let Σ , the **r.e. subobject classifier**, be the closure of the decidable truth-values $2 = \{\top, \perp\} \rightarrow \Omega$ under countable joins; that is,

$$\Sigma = \{p \in \Omega \mid \exists f \in 2^{\mathbf{N}}. p \leftrightarrow (\exists n. f(n) = \top)\}.$$

It's clearly a sublattice of Ω . Observe that by Markov's Principle, the truth-values in Σ are $\neg\neg$ -closed; i.e.

$$\forall p \in \Omega. \neg\neg(p \in \Sigma) \rightarrow p \in \Sigma.$$

So $\Sigma \rightarrow \nabla 2$.

Using Church's Thesis, it's not hard to see that

$$\Sigma \cong \{p \in \Omega \mid \exists e \in \mathbf{N}. p \leftrightarrow \{e\}(e) \downarrow\}.$$

That is, $\Sigma \cong Z_K$ (see 2.5) where K is the halting set; so Σ is actually a modest set.

If the classifying map of a mono $X' \rightarrow X$ factors through $\Sigma \rightarrow \Omega$, we call X' a Σ -subset of X and write $X' \subseteq_{\Sigma} X$.

Σ is a **dominance** in the sense of Rosolini: that is, it has the following properties, emphasized in [69]:

$$\begin{aligned} \top &\in \Sigma; \\ p \in \Sigma \wedge (p = \top \rightarrow q \in \Sigma) &\rightarrow (p \wedge q) \in \Sigma. \end{aligned}$$

This means that the partial maps in $\mathcal{E}ff$ whose domains are Σ -subsets form a category (they are closed under identities and composition).

The following result is true for any dominance Σ .

Proposition 5.1.1 (*cf. [68, 3.2]*) *If $m: X' \rightarrow X$ is a Σ -subset then any map $X' \rightarrow \Sigma$ extends to a map $X \rightarrow \Sigma$; in fact, $\Sigma^m: \Sigma^X \rightarrow \Sigma^{X'}$ is split epi.*

Proof. Define $\Sigma^{X'} \rightarrow \Sigma^X$ by

$$U \mapsto X' \cap U.$$

The second property mentioned above tells us this map is well-defined. ■

Note that if Σ is a modest set in a realizability topos, as in our case, there is an *internal* version of this result, for modest sets; that is, a certain statement of the form

$$\forall R \in \mathbf{Mod}_0. (\forall R' \in \mathbf{Mod}_0. \mathbf{N}/R' \subseteq_{\Sigma} \mathbf{N}/R \rightarrow \exists f: \Sigma^{\mathbf{N}/R'} \rightarrow \Sigma^{\mathbf{N}/R} \dots)$$

is internally valid. It's very long to write out in full (the fragment above is itself an abbreviation) so I won't.

For expository purposes it's often useful to blur the distinction between internal and external. After all, whether we are proving a result about modest sets (in any slice) or about the internal category $\mathbf{Mod}$, the argument is usually the same; in fact they will generally be equivalent—if we are careful how we state them.

5.2 Σ -spaces

Let X be an object of $\mathcal{E}ff$. A map $X \rightarrow \Sigma$ can be regarded as a computational predicate on X . We are interested in those objects X in which two elements which have the same computational properties must be equal. In other words, X has enough Σ -subsets to separate its elements.

Formally, call X a Σ -space if it satisfies

$$\forall x, x' \in X. (\forall f \in \Sigma^X. f(x) = f(x')) \rightarrow x = x'.$$

For example, Σ and $\mathbf{N}$ are Σ -spaces.

It's easy to show that X is a Σ -space if and only if ' ev ': $X \rightarrow \Sigma^X$ is mono (where ' ev ' is the exponential transpose of the evaluation map $ev: X \times \Sigma^X \rightarrow \Sigma$); but the definition using the internal language is more convenient.—Anyway, let Sig be the full subcategory of Σ -spaces.

[Note: in the interests of clarity we will often pretend that an element of Σ^X 'is' a map $X \rightarrow \Sigma$, and so on. It would be more rigorous to write the proofs out formally in the internal language, or categorically using Kripke-Joyal semantics, and some readers may enjoy doing this.]

Proposition 5.2.1 *Sig is closed under subobjects. In fact, an object X is a Σ -space if and only if it's a subobject of some Σ^Y .*

Proof. The first statement is obvious; so we only need to show that Σ^Y is a Σ -space. But for each $y \in Y$, there is a map $\Sigma^Y \rightarrow \Sigma$ that acts on each $U \subset_\Sigma Y$ as follows:

$$U \mapsto |y \in U|$$

(this is just evaluation at y); and the maps of this form suffice to distinguish between elements of Σ^Y . ■

Since Σ is modest and $\mathbf{Mod}$ is closed under subobjects and exponentiation by any object of $\mathcal{E}ff$, every Σ -space is modest. Clearly one can use the internal language to define a small subcategory $\mathbf{Sig}$ of $\mathbf{Mod}$ whose externalization is (weakly equivalent to) Sig ; so Sig is essentially small too.

[We are already being sloppy. The point is that the definition of Σ -space makes sense in any fibre of $\mathcal{E}ff$; so we have really defined a full subfibration Sig of $\mathcal{E}ff$. The relationship between Sig and $\mathbf{Sig}$ is the same as that between $\{\mathbf{Mod}\}$ and $\mathbf{Mod}$: see 2.6. Also, we have not written out the predicate that defines $\mathbf{Sig}_0$ as a subobject of $\mathbf{Mod}_0$ —a $\neg\neg$ -closed subobject, as it turns out—as this would be quite a long formula, involving many statements about PERs.]

Proposition 5.2.2 *Sig is reflective; the reflection of $X \in \mathcal{E}ff$ is the object X_Σ*

appearing in the image factorization of 'ev':

$$\begin{array}{ccc} X & \xrightarrow{\text{'ev'}} & \Sigma^{\Sigma^X} \\ & \searrow \quad \nearrow & \\ & X_{\Sigma} & \end{array}$$

Proof. X_{Σ} is a subobject of Σ^{Σ^X} and hence a Σ -space. Now, suppose that Y is a Σ -space and $f: X \rightarrow Y$; to show that f factors uniquely through X_{Σ} , it suffices to show that f coequalizes the kernel pair of 'ev'. But suppose $x, x' \in X$ and

$$\forall g \in \Sigma^X. g(x) = g(x').$$

Then

$$\forall h \in \Sigma^Y. hf(x) = hf(x')$$

and so, since Y is a Σ -space, $f(x) = f(x')$. ■

As a consequence, *Sig* is complete. Note that since *Mod* and *Sig* are full subcategories of *Eff* containing the terminal object, the limits in these categories are the ordinary 'set-theoretic' ones (i.e. they can be computed in *Eff*): see [32, 2.8]. I should stress again that the word 'complete' has to be interpreted in the right sense (see 2.6); there is a stack *Sig* which is complete as a fibration, and there is an internal category **Sig** which is complete, but not in the strongest possible way... But the problems are the same as those that arose in the case of *Mod*.

Any Σ -space X inherits a partial order from Σ^{Σ^X} , which we call the Σ -order. Explicitly, if $x, x' \in X$ then

$$x \leq x' \text{ if and only if } \forall f \in \Sigma^X. f(x) \leq f(x').$$

(Actually, this definition makes sense for any object of *Eff*, though in general it only defines a *preorder*.)

We think of $x \leq x'$ as saying that ' x' is more defined than x '; for example, we'll see that the object of partial recursive functions $\mathbb{N} \rightarrow \mathbb{N}$ is a Σ -space; and if ψ, ψ' are partial recursive functions then $\psi \leq \psi'$ if and only if ψ' extends ψ . The following fact is trivial but quite important.

Proposition 5.2.3 *If X and Y are Σ -spaces then any map $f: X \rightarrow Y$ is automatically monotone.*

Suppose X is a Σ -space and $X' \rightarrow X$; under what conditions does X' inherit the Σ -order of X , in the sense that

$$\forall x, x' \in A. (x \leq x' \text{ in } X') \leftrightarrow (x \leq x' \text{ in } X) \quad ?$$

The previous remark implies that $\rightarrow$ always holds, but $\leftarrow$ may fail; for example, $2 \rightarrow \Sigma$ but the Σ -order on 2 is discrete.

We call $X' \rightarrow X$ a **subspace** of X if the above condition holds. (This terminology is provisional; probably it will be better to have a stronger, topological definition of subspace.)

Proposition 5.2.4 *If $m: X' \rightarrow X$ induces an epi $\Sigma^m: \Sigma^X \rightarrow \Sigma^{X'}$ then X' is a subspace of X .*

Proof. The condition says that any map $X' \rightarrow \Sigma$ extends to a map $X \rightarrow \Sigma$; so the result is immediate. ■

For example, if $X' \subseteq_{\Sigma} X$ then by 5.1.1, X' is a subspace of X . (A different criterion, that applies only if X is a linked Σ -space, is given later—see 5.4.6.)

Before moving on, I should mention another reason why we can regard Σ -spaces as truly computational in nature. Recall that if X and Y are objects in a cartesian closed category, we call X **orthogonal** to Y if and only if the map $Y \rightarrow 1$ induces an isomorphism $X^Y \cong X$. In a topos, this is the same as saying that (internally) all maps $f: Y \rightarrow X$ are constant. If X is orthogonal to Y we write $X \perp Y$.

In 2.8 it was shown that for any nonrecursive set A , there is an object 2_A in $\mathcal{E}ff$ that ‘classifies A -recursive subobjects’; for example, maps $\mathbb{N} \rightarrow 2_A$ correspond to sets recursive in A (just as maps $\mathbb{N} \rightarrow 2$ correspond to recursive sets).

It’s easy to see that $\Sigma \perp 2_A$ always; for if not, every A -recursive subset would have to be r.e. (see section 2.8 or [50, proposition 2]). And since the category $\text{Orth}(2_A)$ of objects orthogonal to 2_A is closed under subobjects and exponentials (indeed, under all internal limits: [33, section 2]), any Σ -space is in $\text{Orth}(2_A)$. So if we define

$$\mathcal{L} = \bigcap_{A \in \mathcal{R}ef} \text{Orth}(2_A)$$

then $\text{Sig} \subset \mathcal{L}$. That is, ‘ Σ -spaces contain no non-effective information’.

5.3 Chain-complete posets

In this section we prove some (slightly surprising) results about the relationship between chain-complete posets and Σ , that will come in handy soon. The first proposition must have been known to Rosolini; the second certainly appears in [69].

Let $(P, \sqsubseteq)$ be a chain-complete poset in $\mathcal{E}\mathcal{F}$.

Proposition 5.3.1 *Suppose $p, p' \in P$ and $U \subseteq_{\Sigma} P$. If $p \in U$ and $p \sqsubseteq p'$ then $p' \in U$ (Σ -subsets are upward-closed').*

Proof. As in the proof of 5.5.2, or of 5.3.3 in [69], we can define a map $f: \mathbb{N} \rightarrow P$ such that

$$f(n) = \begin{cases} p & n \notin K \\ p' & n \in K \end{cases}$$

(where K is the halting set); that is, for each n , $f(n) = \sup_m g(n, m)$ where

$$g(n, m) = \begin{cases} p & \forall m' \leq m. \neg T(n, n, m') \\ p' & \exists m' \leq m. T(n, n, m') \end{cases}$$

And then $f^{-1}U \subseteq_{\Sigma} \mathbb{N}$, since $U \subseteq_{\Sigma} P$. But if $p' \notin U$, then $f^{-1}U = \bar{K}$, a contradiction; so $\neg p' \notin U$, and so $p' \in U$ since U is $\neg\neg$ -closed. ■

Some of the proofs in chapter 7 use a more abstract version of this argument. It seems to go back to the Rice-Shapiro theorem, but has undergone a number of transformations.

Corollary 5.3.2 (Scott) *If $p \sqsubseteq p'$, then $p \leq p'$.*

Proposition 5.3.3 (Rice-Shapiro-Rosolini) *Suppose (p_n) is a chain in P . If $U \subseteq_{\Sigma} P$ then*

$$\sup_n p_n \in U \leftrightarrow \exists n. p_n \in U$$

(Σ -subsets are Scott-open').

Proposition 5.3.4 (Scott) *If $(P, \sqsubseteq)$ is a chain-complete poset with either binary meets or binary joins, and P is a Σ -space, then $p \sqsubseteq p'$ iff $p \leq p'$.*

Proof. (Johnstone) We must prove that if $p \leq p'$ then $p \sqsubseteq p'$. Assume that $(P, \sqsubseteq)$ has binary joins, denoted by $\sqcup$; the proof for a poset with meets is the same.

Now since the map $- \sqcup p': P \rightarrow P$ is monotone with respect to the Σ -order,

$$p \sqcup p' \leq p' \sqcup p' = p'.$$

But $p' \sqsubseteq p \sqcup p'$, and so $p' \leq p \sqcup p'$. Hence $p' = p \sqcup p'$; that is, $p \sqsubseteq p'$. ■

The hypothesis that P is a Σ -space is essential; for example, Ω satisfies all the other hypotheses, but the Σ -order on it is chaotic.

Corollary 5.3.5 *The Σ -order on Σ coincides with the entailment order; and the Σ -order on any object Σ^X coincides with the inclusion order.*

5.4 Linked Σ -spaces

Suppose $\mathbf{D}$ is an internal category in $\mathcal{E}ff$. Since Sig is complete, for any diagram $F: \mathbf{D} \rightarrow Sig$ there is a limit cone in Sig , say

$$(\pi_d: \varprojlim F \rightarrow F(d))_{d \in \mathbf{D}}.$$

When do the Σ -orders on the $F(d)$ determine the Σ -order on the Σ -space $X = \varprojlim F$? That is, when is it true that

$$\forall x, x' \in X. (x \leq x') \leftrightarrow (\forall d \in \mathbf{D}. \pi_d x \leq \pi_d x') \quad ?$$

Clearly $\rightarrow$ always holds, since each π_d is monotone; if $\leftarrow$ holds, we say that the Σ -order on X is **pointwise** (in the special cases of products and exponentials this just says that X has the pointwise order in the usual sense).

Proposition 5.4.1 *If X and Y are Σ -spaces, then $X \times Y$ has the pointwise Σ -order.*

Proof. (Scott) Suppose $(x, y), (x', y') \in X \times Y$, and suppose $x \leq x'$ in X and $y \leq y'$ in Y ; we want to show that $(x, y) \leq (x', y')$ in $X \times Y$, i.e. that $f(x, y) \leq f(x', y')$ for all $f \in \Sigma^{X \times Y}$. But $f(\cdot, y) \in \Sigma^X$ and $f(x', \cdot) \in \Sigma^Y$ are both monotone, so

$$f(x, y) \leq f(x', y) \leq f(x', y').$$

■

Unfortunately it's not obvious that the limit of any diagram of Σ -spaces should have the pointwise Σ -order; in fact, I suspect that it's false in general. But sometimes it's easy to show that X is ordered pointwise.

Call a Σ -space X **linked** if

$$\forall x, x' \in X. (x \leq x') \rightarrow (\exists f \in X^\Sigma. f(\perp) = x \wedge f(\top) = x').$$

(The converse implication always holds.) We say that f **links** x and x' . This condition is related to m -completeness; for example,

Proposition 5.4.2 *Let A be any nonrecursive r.e. set not m -equivalent to the halting set K . Then the Σ -space Z_A is orthogonal to Σ (and hence Z_A is not linked).*

Proof. Since $\Sigma = Z_K$, if there were a nonconstant map $\Sigma \rightarrow Z_A$ then either $A \leq_m K$ or $\bar{A} \leq_m K$. ■

The same example shows us that the image of a map between linked Σ -spaces may not be a linked Σ -space: for Z_A is the image of the composite $\mathbf{N} \rightarrow \mathbf{N} \rightarrow \Sigma$, where the first map enumerates A .

An important observation is that if X is a linked Σ -space then for any pair $x \leq x'$ in X , the map $f \in X^\Sigma$ linking x and x' is *unique*. This is a consequence of the following simple result.

Lemma 5.4.3 *Let X be a $\neg\neg$ -separated object. Then the map $X^\Sigma \rightarrow X \times X$ induced by evaluating at $\top$ and $\perp$ is mono. That is, any map $f \in X^\Sigma$ is uniquely determined by its values at $\top$ and $\perp$.*

Proof. The maps $\top, \perp: 1 \rightarrow \Sigma$ give us a mono $2 \rightarrow \Sigma$. This mono is easily seen to be $\neg\neg$ -dense; so, since X is $\neg\neg$ -separated, the induced map $X^\Sigma \rightarrow X^2$ is mono. (The argument in this form is due to Peter Johnstone; my original proof was a couple of lines longer, but essentially the same.) ■

So the property of being linked is more uniform than it looks; if X is a linked Σ -space then $X^\Sigma \cong O_X$, where

$$O_X = \{(x, x') \in X \times X \mid x \leq x'\}.$$

An axiom of this sort has been adopted by both Taylor ([77]) and Hyland ([30]) in their work on axiomatic domain theory.

We'll see later that both $\mathbf{N}$ and Σ are linked Σ -spaces; in fact, it seems that any useful Σ -space is linked.

Anyway, the result we're really interested in—the whole reason for defining this category $\mathcal{LSig}$ of linked Σ -spaces—is the following.

Proposition 5.4.4 *Let $F: \mathbf{D} \rightarrow \mathcal{LSig}$ be a diagram of linked Σ -spaces, with limit cone*

$$(\pi_d: \varprojlim F \rightarrow F(d))_{d \in \mathbf{D}}.$$

Then $X = \varprojlim F$ is a linked Σ -space and has the pointwise Σ -order.

Proof. We know X is a Σ -space. Suppose $x, x' \in X$ with $x \leq x'$; then, since each π_d is monotone,

$$\forall d. \pi_d x \leq \pi_d x';$$

and so, since each $F(d)$ is linked,

$$\forall d. \exists f_d \in F(d)^\Sigma. f_d(\perp) = \pi_d x \wedge f_d(\top) = \pi_d x'.$$

But by uniqueness (lemma 5.4.3), the maps $(f_d)_{d \in \mathbf{D}}$ form a cone over F ; so there is a (unique) map $f \in X^\Sigma$ such that $f_d = \pi_d f$, i.e.

$$\forall d. \pi_d f(\perp) = \pi_d x \wedge \pi_d f(\top) = \pi_d x'.$$

Hence $f(\perp) = x$ and $f(\top) = x'$; so X is linked. A similar argument shows that X is ordered pointwise. ■

(For example, if X is a linked Σ -space then for any object Y , the exponential X^Y is a linked Σ -space with the pointwise Σ -order.) So $\mathcal{LSig}$ is complete, and hence reflective; in fact,

Proposition 5.4.5 *If X is a Σ -space then the map $l: X \rightarrow X_L$ from X into its linked reflection is mono, and X is a subspace of X_L .*

Proof. $l: X \rightarrow X_L$ is the initial object of the category whose objects are maps $X \rightarrow Y$ ($Y \in \mathcal{LSig}$) and whose maps are commutative triangles. To see that l is mono, suppose $x, x' \in X$ and $l(x) = l(x')$. Since Σ is linked, any $f \in \Sigma^X$ factors through l ; so

$$\forall f \in \Sigma^X. f(x) = f(x').$$

But X is a Σ -space so $x = x'$. Also, again since Σ is linked, any $f: X \rightarrow \Sigma$ extends to X_L ; so X is a subspace of X_L . ■

In fact, since Σ is linked, $\Sigma^X \cong \Sigma^{X_L}$; it follows from this that X_L is the smallest linked subobject of Σ^X containing X .

If X is a linked Σ -space, then the following propositions imply that a large class of subobjects of X are subspaces. This includes any subobject described using $\leq, =, \neg, \wedge, \perp, \rightarrow, \forall$ —for example,

$$\{x \in X \mid \forall x', x'' \in X. (x' \leq x) \wedge (x'' \leq x) \rightarrow x' = x''\}.$$

Proposition 5.4.6 *If X is linked and $m: X' \rightarrow X$ is $\neg\neg$ -closed then X' is linked and X' is a subspace of X .*

Proof. Suppose we have $x, x' \in X'$ with $x \leq x'$ in X ; then since X is linked,

$$\exists f \in X^\Sigma. f(\perp) = x \wedge f(\top) = x'.$$

Internally speaking, we have a diagram

$$\begin{array}{ccc} 2 & \rightarrow & \Sigma \\ \downarrow & & \downarrow f \\ X' & \rightarrow & X \end{array}.$$

But $2 \rightarrow \Sigma$ is $\neg\neg$ -dense and $X' \rightarrow X$ is $\neg\neg$ -closed, so f factors through X' . ■

The following corollary says that in taking the linked reflection of a Σ -space, we aren't adding any extra elements; intuitively, we only need to add some extra codes for the elements that are already there. This result won't be needed later.

Corollary 5.4.7 *If X is a Σ -space,*

$$\forall x \in X_L. \neg\neg(x \in X);$$

in other words, X is $\neg\neg$ -dense in X_L .

Proof. Let $x \in X_L$, and suppose $\neg(x \in X)$; let

$$X' = \{x \in X_L \mid x \neq x'\} \rightarrow X_L;$$

it is a $\neg\neg$ -closed subobject of X_L and hence itself a linked Σ -space. Now if $x' \in X$ then $x' \neq x$ and so $x' \in X'$; so $X \rightarrow X'$. But X' is strictly smaller than X_L , so this contradicts the fact that X_L is the smallest linked subobject of Σ^X containing X . Therefore $\neg\neg(x \in X)$. ■

Proposition 5.4.8 *For any Σ -space X , the Σ -order is a $\neg\neg$ -closed relation:*

$$\forall x, x' \in X. \neg\neg(x \leq x') \rightarrow (x \leq x').$$

Proof. It's easy to check that entailment is a $\neg\neg$ -closed relation on Σ . Hence it is a $\neg\neg$ -closed relation on any Σ^I , and in particular on Σ^{Σ^X} . But the Σ -order on X is by definition the partial order it inherits from Σ^{Σ^X} . ■

5.4.1 When is the link relation transitive?

Forget about the Σ -order for the moment. If X is any object of $\mathcal{E}ff$, we have a relation $\triangleleft$ on X defined by

$$x \triangleleft x' \iff \exists f \in X^\Sigma. f(\perp) = x \wedge f(\top) = x'$$

(by 5.4.3, if $X \in Sep$ then such an f must be unique). Call this the **link relation**; it is clearly reflexive.

Lemma 5.4.9 *The link relation on the limit of a diagram of $\neg\neg$ -separated objects is pointwise (cf. 5.4.4).*

Proof. Same argument as in 5.4.4. ■

Call an object $X \in Sep$ **link-preordered** if the link relation $\triangleleft$ on X is transitive.

Proposition 5.4.10 *The category of link-preordered objects is complete.*

(That is, the subfibration of $Sep^2 \rightarrow Sep$ consisting of families of link-preordered objects is complete.)

Proof. Let $X = \varprojlim X_i$; suppose we have $x, x', x'' \in X$ with $x \triangleleft x', x' \triangleleft x''$; we want to show that $x \triangleleft x''$.

But if $x = (x_i), x' = (x'_i), x'' = (x''_i)$ then by 5.4.9,

$$x_i \triangleleft x'_i, x'_i \triangleleft x''_i$$

and hence

$$x_i \triangleleft x''_i$$

for all i ; so $x \triangleleft x''$, again by 5.4.9. Hence X is link-preordered. ■

If X is link-preordered, we can define an equivalence relation $\sim$ on X by

$$x \sim x' \leftrightarrow x \triangleleft x' \wedge x' \triangleleft x.$$

Call $X \in \mathbf{Sep}$ link-ordered if $\triangleleft$ is a partial order on X ; that is, if $\sim$ is the equality relation.

Proposition 5.4.11 *The category of link-ordered objects is complete.*

Proof. Let $X = \varprojlim X_i$ and suppose we have $x, x' \in X$ with $x \sim x'$. Then $x_i \sim x'_i$ and hence $x_i = x'_i$ for all i ; so $x = x'$. ■

Proposition 5.4.12 *Any link-ordered object must be modest; so the category of link-ordered objects is small.*

Proof. Recall from [33] that $\mathbf{Mod} \simeq \mathbf{Sep} \cap \mathbf{Orth}(\nabla 2)$; so it suffices to show that any link-ordered object X is orthogonal to $\nabla 2$. But any map $f: \nabla 2 \rightarrow X$ gives rise to two ‘opposite’ maps $\Sigma \rightarrow X$ (arising from the two ‘opposite’ embeddings $\Sigma \hookrightarrow \nabla 2$); so, as X is link-ordered, both of these maps must be constant, and so f must have been constant. ■

Corollary 5.4.13 *The category of link-ordered objects is reflective.*

We prove this using the adjoint functor theorem.

Note. We can’t say that the reflection into the category of link-ordered objects is given by $X \mapsto X/\sim$; there may conceivably be maps $\Sigma \rightarrow X/\sim$ that don’t come from maps $\Sigma \rightarrow X$.

5.4.2 Sketch of a more external approach to completeness

Here we give a different—more ‘categorical’—presentation of the proof that $\mathbf{LSig}$ is complete. It uses the language of fibrations rather than the internal logic of $\mathcal{E}ff$ (though it still relies on a couple of facts proved using the internal logic); this makes it more concrete but less readable.

As in 2.6, the main result is about completeness relative to $\mathbf{Sep}$; everything else can be deduced from this.

Proposition 5.4.14 $\mathcal{LSig}$, thought of as a subfibration of $\text{cod}: \text{Sep}^2 \rightarrow \text{Sep}$, is closed under Π -functors, and finite limits in each fibre.

Note. It is not really a subfibration; rather, there is a full and faithful cartesian functor between the two fibrations.

Proof. First we show that $\mathcal{LSig}$ is closed under finite limits in the fibre over 1 (the argument for other fibres is the same). Let

$$X = \varprojlim X_i$$

be the limit of a finite diagram of linked Σ -spaces; we want to find a map

$$O_X \rightarrow X^\Sigma$$

that's inverse to the map $X^\Sigma \rightarrow O_X$ which sends $f \in X^\Sigma$ to $\langle f(\perp), f(\top) \rangle$.

Since the X_i are linked, we have maps

$$O_{X_i} \rightarrow X_i^\Sigma$$

inverse to the maps $X_i^\Sigma \rightarrow O_{X_i}$. But

$$X^\Sigma \cong \varprojlim X_i^\Sigma.$$

Now the projections $X \rightarrow X_i$ are order-preserving, so they induce maps $O_X \rightarrow O_{X_i}$; hence, by composition, maps

$$O_X \rightarrow O_{X_i} \rightarrow X_i^\Sigma.$$

By 5.4.3, these maps form a cone; so there is an induced map $O_X \rightarrow X^\Sigma$.

As for Π -functors: let

$$\left(\begin{smallmatrix} X \\ I \end{smallmatrix}\right) \in \mathcal{LSig}(I)$$

and let $f: I \rightarrow J$. For typographical reasons we will denote the order relation on $\left(\begin{smallmatrix} X \\ I \end{smallmatrix}\right)$ by

$$O\left(\begin{smallmatrix} X \\ I \end{smallmatrix}\right).$$

We want a map

$$O(\Pi_f\left(\begin{smallmatrix} X \\ I \end{smallmatrix}\right)) \rightarrow (\Pi_f\left(\begin{smallmatrix} X \\ I \end{smallmatrix}\right))^{J^*\Sigma}$$

in the fibre over J .

But $(\Pi_f(\mathcal{X}))^{J^*\Sigma} \cong \Pi_f(\mathcal{X})^{I^*\Sigma}$ and so, since $f^* \dashv \Pi_f$, it suffices to find a suitable map

$$f^*O(\Pi_f(\mathcal{X})) \rightarrow (\mathcal{X})^{I^*\Sigma}.$$

But since the I -indexed product projection

$$f^*\Pi_f(\mathcal{X}) \rightarrow (\mathcal{X})$$

is monotone, it induces a map

$$f^*O(\Pi_f(\mathcal{X})) \rightarrow O(\mathcal{X});$$

and since $(\mathcal{X})$ is an I -indexed family of linked Σ -spaces, by definition we have a map

$$O(\mathcal{X}) \rightarrow (\mathcal{X})^{I^*\Sigma};$$

so again, we just have to compose the two to get a map with the desired properties.

■

Perhaps we should also restate the result that ‘the Σ -order on a limit of linked Σ -spaces is pointwise’:

Corollary 5.4.15 *Let (X_i) be a diagram of linked Σ -spaces and let $X = \varprojlim X_i$. Then*

$$O_X = \varprojlim O_{X_i}.$$

See sub-section 5.5.1 for more on **LSig**.

5.5 Complete Σ -spaces

Now we come to the category we’re really interested in. Call a Σ -space X **complete** if it has sups of $\mathbb{N}$ -chains under the Σ -order. For example, $\mathbb{N}$ is complete since the Σ -order on $\mathbb{N}$ is discrete; and Σ is complete by 5.3.5.

Let **CSig** be the full subcategory of complete Σ -spaces; it will be our category of ‘predomains’.

Since sups of $\mathbb{N}$ -chains are unique, we see that if X is a Σ -space and $C_X \rightarrowtail X^{\mathbb{N}}$ is the object of chains in X —that is,

$$C_X = \{f \in X^{\mathbb{N}} \mid \forall n. f(n) \leq f(n+1)\}$$

—then X is a complete Σ -space if and only if there is a map $C_X \rightarrow X$ sending each chain to its sup. (From now on, we'll use the notation (x_n) for a chain $\mathbb{N} \rightarrow X$.)

The following result is what makes everything work. It looks quite surprising at first, but its proof is remarkable simple.

Proposition 5.5.1 *Let X be a complete Σ -space and let Y be any Σ -space. Then any map $f: X \rightarrow Y$ preserves sups of chains.*

Proof. Let (x_n) be a chain in X , and let $x = \sup_n x_n$.

Since f is monotone, $(f(x_n))$ is a chain in Y ; to show that $f(x) = \sup_n f(x_n)$, it suffices to show that

$$\forall y \in Y. (\forall n. y \geq f(x_n)) \rightarrow y \geq f(x).$$

So suppose we have $y \in Y$ with $y \geq f(x_n)$ for all $n \in \mathbb{N}$; we must show that for any $U \subset_\Sigma Y$, if $f(x) \in U$ then $y \in U$.

Suppose then that $f(x) \in U$. Then $x \in f^{-1}(U) \subset_\Sigma X$; but $x = \sup_n x_n$ so (by 5.3.3) for some n , $x_n \in f^{-1}(U)$. Then $f(x_n) \in U$ and so, since $f(x_n) \leq y$, we have $y \in U$. ■

(Note that this result is very closely related to the Rice-Shapiro and Myhill-Shepherdson theorems. For example, part of the Rice-Shapiro theorem is the statement that any map $\Sigma^\mathbb{N} \rightarrow \Sigma$ is determined by what it does to the finite subsets; since any r.e. set is the union of a chain of finite subsets, this follows from the above proposition. A better discussion of the Rice-Shapiro theorem in *Eff* appears in [68].)

The following result shows that limits of complete Σ -spaces have the pointwise order; it also shows that, despite appearances, the object Z_A (where $\phi \preceq_m A \preceq_m K$) is not complete; ‘sups of chains in Z_A are not computable’. This example reminds us that we can’t just look at the partial order in an external way; we must work in the internal logic to avoid being led astray.

Proposition 5.5.2 *Any complete Σ -space X is linked.*

Proof. Suppose $x, x' \in X$ and $x \leq x'$. For each $n \in \mathbb{N}$ define a chain in X as follows:

$$x_{mn} = \begin{cases} x & \forall m' \leq m. \neg T(n, n, m') \\ x' & \exists m' \leq m. T(n, n, m'). \end{cases}$$

(Definition by cases makes sense since the T -predicate is decidable.) Since X is complete, we can define

$$x_n = \sup_m x_{mn} = \begin{cases} x & \{n\}(n) \uparrow \\ x' & \{n\}(n) \downarrow. \end{cases}$$

So we have a map $\mathbf{N} \rightarrow X$ that coequalizes the kernel pair of $\mathbf{N} \rightarrow \Sigma$ (and in fact this map is uniquely determined); hence we get a map $\Sigma \rightarrow X$ as required. (Note that this proof can be formalized in the internal logic.) ■

There is a more abstract version of this proof that uses the fact that Σ is the closure of 2 (the lattice of decidable truth-values) under countable unions; see the discussion in [74], and section 7.4.

Corollary 5.5.3 *$\mathcal{CSig}$ is complete.*

Proof. Let $X = \varinjlim F$ be the limit of a diagram $F: \mathbf{D} \rightarrow \mathbf{CSig}$, and let $\pi_d: X \rightarrow F(d)$ be the projections. We know that X is linked and has the pointwise Σ -order.

Given a chain (x_n) in X , we have chains $(\pi_d x_n)$ in each $F(d)$; let

$$x_d = \sup_n \pi_d x_n.$$

Since each $F(d)$ is a complete Σ -space, each map in the diagram preserves sups of chains; that is, for any $f: d \rightarrow d'$ in $\mathbf{D}$ we have

$$F(f)(x_d) = x_{d'}.$$

So there is a unique $x \in X$ such that $\pi_d x = x_d$; and since X has the pointwise order, it's easy to see that $x = \sup_n x_n$. ■

There is also a more 'external' version of this proof, just as there was for the result involving linked Σ -spaces (see the discussion at the end of the previous section).

One consequence of this is that many domain equations can be solved inside $\mathcal{CSig}$ —namely the 'covariant' ones which can be solved by finding an initial algebra $TX \rightarrow X$ for a functor $\mathbf{CSig} \rightarrow \mathbf{CSig}$ (cf. [28]). Since the category is complete, initial algebras always exist.

Corollary 5.5.4 *$\mathcal{CSig}$ is reflective.*

Proof. Again, by the adjoint functor theorem; see 2.7. ■

It's frustrating that there's no explicit description of the chain completion X_C ; the obvious candidates don't seem to work. To prove anything about X_C , we have to rely on the fact that $\Sigma^X \cong \Sigma^{X_C}$. For example:

Proposition 5.5.5 *If X is a Σ -space then the map $X \rightarrow X_C$ from X into its reflection in $\mathcal{CSig}$ is mono, and X is a subspace of X_C .*

Proof. Both follow from the fact that Σ is a complete Σ -space, so that any map $X \rightarrow \Sigma$ factors uniquely through X_C ; cf. 5.4.5. ■

A $\neg\neg$ -closed subobject of a complete Σ -space need not be complete. For example, define the object of (classically) finite r.e. subsets of $\mathbb{N}$:

$$Fin = \{W \in \Sigma^{\mathbb{N}} \mid \neg\neg\exists n. m \in W \rightarrow m \leq n\} ;$$

it is a $\neg\neg$ -closed subobject of $\Sigma^{\mathbb{N}}$ and hence a subspace of $\Sigma^{\mathbb{N}}$, but it is certainly not a complete Σ -space.

However, $\mathcal{CSig}$ is closed under retracts (in fact, so are $\mathcal{LSig}$ and $\mathcal{CFSig}$, the category of complete focal Σ -spaces to be defined later—these proofs are left to the reader):

Proposition 5.5.6 *If X is a complete Σ -space, so is any retract X' of X .*

Proof. Suppose we are given $i: X' \rightarrow X$ and $p: X \rightarrow X'$ such that $pi = 1_{X'}$. Let (x_n) be a chain in X' ; find $x = \sup_n x_n$ in X . Then since p preserves sups of chains, $px = \sup_n pi x_n = \sup_n x_n$ in X' . ■

It's also easy to see that a Σ -subset X' of a complete Σ -space X is a complete Σ -space, and a subspace of X : it's upward-closed.

5.5.1 Note on the category $\mathcal{CSig}$

Now we'll step back to take a closer look at $\mathcal{CSig}$ (and $\mathcal{CSig}$). The following remarks are equally applicable to $\mathcal{LSig}$ (just as the remarks in sub-section 5.4.2 are applicable to $\mathcal{CSig}$).

The predicate that classifies the mono

$$\mathbf{Sig}_0 \mapsto \mathbf{Mod}_0$$

—namely, the property of being a Σ -space—is $\neg\neg$ -stable, and hence this mono is $\neg\neg$ -closed. (So Sig_0 is a $\neg\neg$ -sheaf.) One could say that the objects of Sig are simply PERs, and being a Σ -space in $\mathcal{E}ff$ means the same thing as being a ‘ Σ -space’ in $Sets$.

On the other hand,

$$CSig_0 \rightarrow Sig_0$$

is not $\neg\neg$ -closed; the property of being a complete Σ -space is not $\neg\neg$ -stable, because the defining predicate contains an existential quantifier. We think of an object of $CSig$ as being a PER R *together with* a map

$$\sup: C_{\mathbb{N}/R} \rightarrow \mathbb{N}/R$$

taking chains to their sups. That is, being complete is stronger than just being a ‘ Σ -space with sups of chains’ in $Sets$: we have this effective procedure that calculates sups. (Elements of non- $\neg\neg$ -closed subobjects can generally be thought of as elements equipped with some extra intensional information.)

But internally speaking, we don’t have to think of a complete Σ -space as being a Σ -space together with extra structure; a complete Σ -space is just a Σ -space satisfying a particular property, from which it’s possible to deduce the existence of a certain map—and we’re free to exploit this fact or simply to ignore it. (Just as in $Sets$, we think of a chain-complete poset as a poset having sups of chains rather than as a poset equipped with a supremum map.)

We have to be more careful when defining the fibration $CSig$; in this case the extra information is not subsumed in an internal logic, and we do have to carry it around explicitly.

That is: if $(\begin{smallmatrix} X \\ I \end{smallmatrix}) \in Sig(I)$ is an I -indexed family of Σ -spaces, let

$$C(\begin{smallmatrix} X \\ I \end{smallmatrix})$$

denote the object of I -indexed families of chains; an element of it is a collection consisting of a chain (x_{i_n}) in each $X_i = p^{-1}(i)$, ‘uniform in I ’. Then an object of $CSig(I)$ is a family $(\begin{smallmatrix} X \\ I \end{smallmatrix})$ *together with* a map

$$C(\begin{smallmatrix} X \\ I \end{smallmatrix}) \rightarrow (\begin{smallmatrix} X \\ I \end{smallmatrix})$$

in $\mathcal{E}ff/I$ that is a ‘uniform procedure for calculating the suprema of a uniformly given family of chains’ (such a map is necessarily unique, of course). However, maps in $CSig$ are still just maps in $\mathcal{E}ff/I$ between the underlying families $(\begin{smallmatrix} X \\ I \end{smallmatrix})$.

Note. This is not the same as saying, externally, that each X_i is an object of $\mathcal{CSig}_i$ for we are only concerned with ‘uniformly given’ families of chains, but on the other hand we insist on there being a uniform way of computing the supremum of each member of such a family. (‘Uniform’ means ‘uniform in I ’: that is, given a code for $i \in I$ we can compute a code for a supremum map for X_i . If I is a constant set, i.e. in the image of $\mathbf{Sets} \hookrightarrow \mathbf{Eff}$, this means there must be a single code that works for all the X_i .)

One other thing about $\mathbf{CSig}$. Since there is a reflection $\mathbf{Sig} \rightarrow \mathbf{CSig}$, we might be tempted to conclude that there is a (split) epi $\mathbf{Sig}_0 \rightarrow \mathbf{CSig}_0$. But we can’t assert this straight away as the composite

$$\mathbf{CSig} \hookrightarrow \mathbf{Sig} \rightarrow \mathbf{CSig}$$

is not necessarily equal to the identity, just naturally isomorphic to it. (Maybe we can make it work if we are more careful about the precise definition of the reflection.)

5.6 Flat Σ -spaces

We call a Σ -space X *flat* if the Σ -order on X is discrete. (The intuition is that every element $x \in X$ is ‘totally defined’: no element of X ‘extends’ x .) Let $\mathcal{FSig}$ be the full subcategory of flat Σ -spaces.

Each flat Σ -space is chain-complete for trivial reasons (there are no nonconstant chains); so, since limits of complete Σ -spaces have the pointwise order, $\mathcal{FSig}$ is complete.

Martin Hyland has pointed out that there are two other candidates for the category of ‘total objects’. One is the category of Δ -spaces, i.e. those objects X with enough decidable subobjects to separate their elements:

$$\forall x, x' \in X. (\forall f \in 2^X. f(x) = f(x')) \rightarrow x = x'.$$

The second is the category $\mathbf{Orth}(\Sigma)$ of objects orthogonal to Σ . Both categories are complete and essentially small ($\mathbf{Orth}(\Sigma)$ is essentially small because it’s a subcategory of $\mathbf{Orth}(\nabla 2)$).

The following propositions set out the relationship between these three categories (and refutes Hyland’s conjecture that every object orthogonal to Σ is a Δ -space). The first one is the important one.

Proposition 5.6.1 *A linked Σ -space is flat if and only if it's orthogonal to Σ .*

Proof. Easy. ■

Proposition 5.6.2 *It can be shown that*

- (i) *any Δ -space is a flat Σ -space (hence linked, and hence orthogonal to Σ);*
- (ii) *not every Σ -space orthogonal to Σ is flat;*
- (iii) *not every flat Σ -space is a Δ -space.*

Proof. (i) is easy. For (ii), consider the object Z_A where $\phi \leq_m A \leq_m K$; it's orthogonal to Σ , but inside it, $\perp \leq \top$.

(iii): Let A and B be disjoint r.e. recursively inseparable sets, and let X be the modest set with underlying set $\{0, 1\}$ and

$$|0| = A, \quad |1| = B.$$

Then the closed subobjects with underlying sets $\{0\}$ and $\{1\}$ are Σ -subsets of X , and so X is flat; but no recursive set can separate the elements of X . ■

Observe that $\mathcal{FSig}$ is closed under arbitrary subobjects, unlike $\mathcal{LSig}$ and $\mathcal{CSig}$. $\mathbf{N}$ is flat, so since $\mathcal{FSig}$ is closed under exponentiation, $\mathbf{N}^{\mathbf{N}}$ (and all the other finite types) are flat; this is what we might hope. In fact, these objects are clearly all Δ -spaces. It seems likely that all the useful flat Σ -spaces are actually Δ -spaces.

[Note: observe that Z_A , 2 , Σ , 2_A and the flat space constructed in the proof of (iii) all have two global sections; which just goes to show, yet again, that global sections rarely tell you much about an object of $\mathcal{E}ff$.]

Chapter 6

Some miscellaneous results

6.1 Chain completeness and orthogonality

The fact that all our categories are *full* subcategories of $\mathcal{E}ff$ suggest that we should be able to do domain theory without partial orders: there should be a purely synthetic, categorical theory. Here we make a start by giving a categorical characterization of chain completeness (of a kind suggested by Hyland) and construct a ‘fixed point classifier’ (closely related to the ‘objects of recursion’ studied in [15]). There are no deep results here though.

The ideas (I think) go back to discussions between Freyd and Hyland some years ago. They have been explored much more fully by other people: see, for example, [77] or [20].

First we construct a ‘generic chain’. Consider the category whose objects are chains $\mathbb{N} \rightarrow X$ where X is a linked Σ -space, and whose morphisms are commutative triangles. If $c: \mathbb{N} \rightarrow \omega$ is the initial object of this category, then any chain in a linked Σ -space factors uniquely through c .

Let $i: \omega \rightarrow \bar{\omega}$ be the inclusion of ω into its chain completion; $ic: \mathbb{N} \rightarrow \bar{\omega}$ is a ‘generic chain with sup’: every chain in a complete Σ -space factors uniquely through ic . Then obviously

Proposition 6.1.1 *Let X be a linked Σ -space. Then X is complete if and only if any map $\omega \rightarrow X$ factors uniquely through i .*

This says that a linked Σ -space X is complete if and only if it’s orthogonal to the map $\omega \rightarrow \bar{\omega}$. (For recursion-theoretic reasons, one wouldn’t expect a similar characterization that worked for arbitrary Σ -spaces.)

Now, $\bar{\omega}$ has a bottom and a top element:

$$\perp = ic(0), \quad \top = \sup_n ic(n).$$

(So if the chain $f: \mathbf{N} \rightarrow X$ factors uniquely as $f = f_c ic: \mathbf{N} \rightarrow \bar{\omega} \rightarrow X$ then $f_c(\top) = \sup_n f(n)$.)

Define a map $s: \bar{\omega} \rightarrow \bar{\omega}$ by

$$sic(n) = ic(n+1).$$

Then clearly $\top$ is the least fixed point of s ; and s has the following universal property ($\top$ is the 'generic least fixed point'):

Proposition 6.1.2 *Let X be a complete focal Σ -space and let $f: X \rightarrow X$. Then there is a unique map $g: \bar{\omega} \rightarrow X$ such that*

$$\begin{array}{ccc} \bar{\omega} & \xrightarrow{s} & \bar{\omega} \\ g \downarrow & & \downarrow g \\ X & \xrightarrow{f} & X, \end{array}$$

$g(\perp) = \perp_X$ and $g(\top)$ is the least fixed point of f .

Proof. Define g by

$$\begin{aligned} gic(0) &= \perp_X \\ gic(n+1) &= fgic(n). \end{aligned}$$

■

If X is a complete focal Σ -space and $f: X \rightarrow X$, let the least fixed point of f be denoted by Υf (we in fact have a map $\Upsilon: X^X \rightarrow X$). Then this universal property of s gives a purely categorical proof of the following well-known (and already trivial) result. Its significance in domain theory was first pointed out by Plotkin.

Corollary 6.1.3 *If $X, Y \in \mathcal{CF}\mathcal{Sig}$ and $f: X \rightarrow X$, $g: Y \rightarrow Y$ and $h: X \rightarrow Y$ satisfy*

$$\begin{array}{ccc} X & \xrightarrow{f} & X \\ h \downarrow & & \downarrow h \\ Y & \xrightarrow{g} & Y \end{array}$$

and if h is strict, then $h(\Upsilon f) = \Upsilon g$.

Also, let $s_{\perp}:\bar{\omega}_{\perp} \rightarrow \bar{\omega}$ be the unique strict extension of s that exists by 7.4.2. (Note that $s_{\perp}$ is *not* the structure map displaying $\bar{\omega}$ as an algebra for the lift monad on $\mathcal{CSig}$.) Then

Proposition 6.1.4 $s_{\perp}:\bar{\omega}_{\perp} \rightarrow \bar{\omega}$ is an isomorphism; in fact, this is the initial solution (in $\mathcal{CSig}$) of the domain equation $X \cong X_{\perp}$.

Remark. The following explicit description of $\bar{\omega}$ may be of interest; it is similar to one found independently by Rosolini and Streicher, who have been studying the category $\text{Orth}(\omega \mapsto \bar{\omega})$. Consider the partial recursive function ϕ defined by $\phi(e) = \{e\}(e)$. Let

$$\begin{aligned} K_0 &= \{n \mid \phi(n) \uparrow\} \\ K_i &= \{n \mid \phi^i(n) \downarrow \text{ but } \phi^{i+1}(n) \uparrow\} \\ K_{\infty} &= \{n \mid \phi^i(n) \downarrow \text{ for all } i\} \end{aligned}$$

That is,

$$\begin{aligned} K_0 &= \{e \mid \{e\}(e) \uparrow\} \\ K_1 &= \{e \mid \{e\}(e) \downarrow, \{\{e\}(e)\}(\{e\}(e)) \uparrow\} \end{aligned}$$

and so on. Let $\bar{\omega} \cong \mathbb{N}/R$ where R is the total equivalence relation

$$nRm \iff \forall 0 \leq i \leq \infty. n \in K_i \leftrightarrow m \in K_i.$$

$\bar{\omega}$ can be thought of as the modest set with underlying set $\{0, 1, 2, \dots, \infty\}$ and

$$|0| = K_0, \dots, |i| = K_i, \dots, |\infty| = K_{\infty}.$$

The proof that any chain in a complete Σ -space factors through this object is similar to that of 5.5.2.

Warning. For linked Σ -spaces chain completeness is equivalent to being orthogonal to $\omega \mapsto \bar{\omega}$; but this is not true for general Σ -spaces. For example, the object Z_A considered in 5.4.2 is orthogonal to $\omega \hookrightarrow \bar{\omega}$ (because it's in fact orthogonal to ω); but it's certainly not chain complete—not even linked.

6.1.1 Not $\neg\neg$ -separated domains?

All the classes of objects studied in chapter 5 lived inside $\mathcal{Sep}$. However, there is no reason why an object orthogonal to $\omega \mapsto \bar{\omega}$ should be $\neg\neg$ -separated. So this suggests the possibility of looking for domains that do not lie in $\mathcal{Sep}$, but

have a nontrivial equality relation (a kind of intensional or observed equivalence perhaps).

We won't investigate this possibility. It might turn out to be a red herring; or it might be more interesting to see how it works in the toposes studied in chapters 12 and 13 rather than in $\mathcal{E}ff$.

6.2 A limit-colimit coincidence

Eventually we will want to solve domain equations. In some cases the solutions are initial algebras of functors; in others, we still have to use the inverse limit method of [62]. With this second possibility in mind, we prove here a familiar result which seems to have been suggested first by Bill Lawvere (though elaborated first by Dana Scott, who realised its significance). Note that the proof, too, looks very familiar.

Proposition 6.2.1 *Suppose we have a sequence of injection-projection pairs in $CSig$ —that is, a family of objects $(X_n)_{n \in \mathbb{N}}$ and families of maps $i_n: X_n \rightarrow X_{n+1}$ and $p_n: X_{n+1} \rightarrow X_n$ satisfying $p_n i_n = 1$, $i_n p_n \leq 1$. Let $X = \varprojlim (X_n, p_n)$. Then $X \cong \varinjlim (X_n, i_n)$, where the colimit is computed in $CSig$.*

Proof. Limits in $CSig$ are the same as the ‘set-theoretic’ ones in $\mathcal{E}ff$; in particular, X is the object of sequences $x = (x_1, x_2, x_3, \dots)$ where $x_n \in X_n$ and $p_n x_{n+1} = x_n$; its Σ -order is the same as the componentwise Σ -order. The projections $\pi_n: X \rightarrow X_n$ defined by $\pi_n(x) = x_n$ form the limiting cone.

Define $\iota_n: X_n \rightarrow X$ by

$$\iota_n(x_n) = (\dots, p_{n-2}p_{n-1}x_n, p_{n-1}x_n, x_n, i_n x_n, i_{n+1}i_n x_n, \dots).$$

Then clearly $\pi_n \iota_n = 1$; and $\iota_n \pi_n \leq 1$, since if

$$x = (\dots, x_{n-1} = p_{n-1}x_n, x_n = p_n x_{n+1}, x_{n+1}, \dots)$$

then

$$\iota_n \pi_n x = (\dots, p_{n-1}x_n, x_n, i_n p_n x_{n+1}, \dots)$$

and so $x \geq \iota_n \pi_n x$ componentwise.

Furthermore, $x = \sup_n \iota_n \pi_n x$; for suppose $x' \geq \iota_n \pi_n x$ for all n . If $x' = (x'_1, x'_2, \dots)$ then by monotonicity of π_n ,

$$x'_n = \pi_n x' \geq \pi_n \iota_n \pi_n x = x_n$$

for all n ; and so $x' \geq x$.

We will show that the $\iota_n: X_n \rightarrow X$ form a colimiting cone. Suppose Y is a complete Σ -space and we have a cocone $f_n: X_n \rightarrow Y$ (that is, $f_n = f_{n+1}i_n$); we want to show that there is a unique map $f: X \rightarrow Y$ such that $f_n = f\iota_n$.

Given $x \in X$, observe that $(f_n\pi_n x)_{n \in \mathbb{N}}$ is a chain in Y ; for if $x = (x_1, x_2, \dots)$ then

$$\begin{aligned} f_n\pi_n x &= f_n x_n = f_n p_n x_{n+1} = f_{n+1} i_n p_n x_{n+1} \\ &\leq f_{n+1} x_{n+1} = f_{n+1} \pi_{n+1} x. \end{aligned}$$

(Remember that $i_n p_n \leq 1$ and that f_{n+1} is monotone).

So we can define

$$f(x) = \sup_n f_n \pi_n x.$$

Clearly $f_n = f\iota_n$, since if $x = \iota_n x_n$ for some $x_n \in X_n$, then the chain $(f_n \pi_n x)_{n \in \mathbb{N}}$ stabilizes by the n th position. And f is the unique such map, since any $x \in X$ is the supremum of $\iota_n \pi_n x$, and f must preserve sups of chains. ■

The result is *not* true if we compute the colimit in $\mathcal{E}ff$ or $\mathcal{S}ig$; the colimit in both cases is the same, and contains only the finite elements.

While we're on the subject of domain equations: Rosolini has recently proved that under appropriate circumstances, *all* (internal) functors preserve limits of sequences of projections. This should be seen as another 'parametricity' result that follows from the fact that such functors must be 'uniformly defined'. We can ensure this for functors on complete Σ -spaces by passing from $\mathbf{CSig}$ to $\overline{\mathbf{CSig}}$ as described in section 3.3.

6.3 All functors are enriched

Complete Σ -spaces have an intrinsic partial order; and all maps between complete Σ -spaces are monotone and preserve sups of chains. That is, all maps respect the intrinsic structure.

What about a functor $F: \mathbf{CSig} \rightarrow \mathbf{CSig}$? Does it have to respect the intrinsic structure too, in the sense that

- if $f, g: X \rightarrow Y \in \mathbf{CSig}$ and $f \leq g$ then $Ff \leq Fg$; and
- if $f_n: X \rightarrow Y$ is a chain of maps then $F(\sup_n f_n) = \sup_n Ff_n$?

(Since Y is linked, we can write $f \leq g$ without ambiguity; it could mean either the pointwise order or the Σ -order in Y^X , and both are the same.) Note that the first condition must hold for the second one to make sense.

Well, yes; so long as F is an *internal* functor. In fact, the result is simple.

Proposition 6.3.1 *Suppose we have a functor $F: \mathbf{LSig} \rightarrow \mathbf{LSig}$ (which induces a functor $F: \mathcal{LSig} \rightarrow \mathcal{LSig}$); and let $f, g: X \rightarrow Y \in \mathcal{LSig}$. If $f \leq g$ then $Ff \leq Fg$.*

Proof. Since F is a functor defined internally, there is a map

$$Y^X \rightarrow FY^{FX}$$

(really a certain map in the fibre over $\mathbf{LSig}_0 \times \mathbf{LSig}_0$) that ‘internalizes’ the action of F . But this map must be monotone with respect to the Σ -order. ■

Proposition 6.3.2 *Suppose we have a functor $F: \mathbf{CSig} \rightarrow \mathbf{CSig}$ (which induces a functor $F: \mathcal{CSig} \rightarrow \mathcal{CSig}$); and let $f_n: X \rightarrow Y \in \mathcal{CSig}$ be a chain of maps. Then $F(\sup_n f_n) = \sup_n Ff_n$.*

Proof. Much the same. ■

It’s possible to generalize this to functors

$$F: \mathbf{CSig}^n \times (\mathbf{CSig}^{op})^m \rightarrow \mathbf{CSig},$$

but we don’t bother to write out the details.

6.4 Lax limits

Since there is an intrinsic partial order on the maps $X \rightarrow Y$ between Σ -spaces, it’s natural to try and investigate the 2-categorical properties of the categories we’ve been studying.

Let $\mathbf{D}$ be a category in $\mathcal{E}ff$. A **lax diagram** of type $\mathbf{D}$ in $\mathbf{Sig}$ consists of maps

$$F_0: \mathbf{D}_0 \rightarrow \mathbf{Sig}_0$$

$$F_1: \mathbf{D}_1 \rightarrow \mathbf{Sig}_1 \quad (\text{respecting domains and codomains})$$

—we’ll just write $d \mapsto Fd$, $(f: d \rightarrow d') \mapsto (Ff: Fd \rightarrow Fd')$ —such that

$$F1_d \leq 1_{Fd}$$

$$F(gf) \leq (Fg)(Ff).$$

A lax cone over F with vertex X is a collection of maps $(p_d: X \rightarrow Fd)_{d \in \mathbf{D}_0}$ such that

$$p_{d'} \leq (Ff)p_d$$

whenever $f: d \rightarrow d' \in \mathbf{D}$. (Note that since everything is partially ordered, there are no coherence conditions to worry about.)

The lax limit of F (if it exists) is the universal lax cone

$$(\pi_d: L \rightarrow Fd)_{d \in \mathbf{D}_0}$$

over F .

For example, let $\mathbf{D} = 1$; then a lax diagram of type D consists of an object Y and a map $h: Y \rightarrow Y$ such that

$$h \leq 1, \quad h \leq h^2$$

(hence $h = h^2$ since h is monotone); its lax limit should be

$$\{y \in Y \mid hy = y\}$$

—that is, we're splitting a decreasing idempotent.

To construct lax limits in the obvious way, we're clearly going to want the Σ -order to be well-behaved in the limit; so we can't expect results for Σ -spaces in general.

Proposition 6.4.1 *$\mathcal{LSig}$ has all (small) lax limits.*

Proof. Let F be a lax diagram of linked Σ -spaces (as above) and define $L \mapsto \Pi_d Fd$ by

$$L = \{(x_d) \mid \forall f: d \rightarrow d'. x_{d'} \leq Ff(x_d)\}.$$

L is a $\neg\neg$ -closed subobject of $\Pi_d Fd$, so by 5.4.6 it is a linked subspace. For each d we have a projection map

$$\pi_d: L \rightarrow Fd.$$

Now, given a lax cone $(p_d: Y \rightarrow Fd)$, the p_d induce a map $Y \rightarrow \Pi_d Fd$; and since for all $y \in Y$ we have

$$\forall f: d \rightarrow d'. p_{d'} y \leq Ff(p_d y),$$

this map factors (uniquely) through L . ■

Proposition 6.4.2 *CSig has lax limits.*

Proof. In the notation above, suppose that all the Fd are complete (so that $\Pi_d Fd$ is complete too); we have to show that L is closed under sups of chains.

Since the Σ -order on L is pointwise, a chain in L must have the form

$$((x_{dn})_{d \in \mathbf{D}_0})_{n \in \mathbf{N}}$$

where

$$\begin{aligned} & \forall d, n. x_{dn} \leq x_{d(n+1)}, \\ & \forall f: d \rightarrow d', n. x_{d'n} \leq Ff(x_{dn}). \end{aligned}$$

Let $x_d = \sup_n x_{dn}$; then $(x_d) \in \Pi_d Fd$.

We just have to prove that $(x_d) \in L$; that is, that

$$\forall f: d \rightarrow d'. x_{d'} \leq Ff(x_d).$$

But (remembering that Ff preserves sups of chains)

$$\begin{aligned} x_{d'} &= \sup_n x_{d'n} \\ &\leq \sup_n Ff(x_{dn}) \\ &= Ff(\sup_n x_{dn}) \\ &= Ff(x_d). \end{aligned}$$

■

Note that the proofs are routine (though we've been pretty informal).

6.5 Weighted limits

$\mathcal{CSig}$ is definitely an enriched category. But enriched over what? (And there are also foundational problems—to do with integrating the theories of enriched categories and fibrations—which we are going to ignore in the following sketchy account.)

Let $\mathbf{Ord}$ be the category of posets in $\mathcal{E}\mathbf{ff}$. Then $\mathcal{LSig}$ and $\mathcal{CSig}$ are $\mathbf{Ord}$ -categories; so of course one wants to know what sort of weighted limits they have (see [41]).

But it turns out to be more natural to regard them as being $\mathcal{LSig}$ -categories; and this is what we are doing for the purposes of the next two results.

Proposition 6.5.1 *$\mathcal{LSig}$ has all (small) weighted limits.*

Proof. Again, this is straightforward. Let $\mathbf{D}$ be a category in $\mathcal{E}ff$, $F: \mathbf{D} \rightarrow \mathbf{LSig}$ a weight.

If $G: \mathbf{D} \rightarrow \mathbf{LSig}$ is a diagram in $\mathbf{LSig}$ of type $\mathbf{D}$, the weighted limit $\{F, G\}$ of G with weight F has the universal property

$$\mathrm{Hom}_{\mathbf{LSig}}(X, \{F, G\}) \cong \mathrm{Nat}_{\mathbf{D}, \mathbf{LSig}}(F, \mathbf{LSig}(X, G-)).$$

Now a natural transformation $(Fd \rightarrow \mathbf{LSig}(X, Gd))_{d \in \mathbf{D}_0}$ is a compatible family of maps

$$Fd \rightarrow Gd^X$$

and so corresponds to a compatible family of maps

$$X \rightarrow Gd^{Fd}.$$

[For example, if F is the constant functor at Σ this means we have, in effect, a compatible family of pairs of maps

$$f, g: X \rightarrow Gd$$

with $f \leq g$; or if F is the constant functor at the object ω defined in 6.1, then we have a compatible family of *chains* of maps

$$(f_n: X \rightarrow Gd)_{n \in \mathbb{N}}. \quad]$$

The limiting ‘cone’ $(\{F, G\} \rightarrow Gd^{Fd})_{d \in \mathbf{D}_0}$ would be the universal such family. Intuitively, $\{F, G\}$ should be the set of families $(\mu_d)_{d \in \mathbf{D}_0}$ (where $\mu_d \in Gd^{Fd}$) ‘that are the components of a natural transformation’.

More formally, put

$$\{F, G\} = \{(\mu_d) \in \prod_d Gd^{Fd} \mid \forall f: d \rightarrow d' \in \mathbf{D}. (Gf)\mu_d = \mu_{d'}(Ff)\}.$$

Then $\{F, G\} \hookrightarrow \prod_d Gd^{Fd}$ is $\neg\neg$ -closed and hence a linked subspace; and the verification that it has the correct universal property is routine. $\blacksquare$

Proposition 6.5.2 *CSig has all weighted limits too.*

Proof. As before, we just have to check that if all the Gd are complete Σ -spaces then the object $\{F, G\}$ we constructed is closed under sups of chains.

But a chain in $\{F, G\}$ is a family of chains $((\mu_{dn})_{n \in \mathbb{N}})_{d \in \mathbf{D}_0}$ that satisfies

$$\forall n \forall f: d \rightarrow d'. (Gf)\mu_{dn} = \mu_{d'n}(Ff);$$

let $\mu_d = \sup_n \mu_{dn}$ —then $(\mu_d) \in \Pi_d Gd^{F^d}$.

But now

$$\begin{aligned} (Gf)\mu_d &= (Gf)\sup_n \mu_{dn} \\ &= \sup_n (Gf)\mu_{dn} \\ &= \sup_n \mu_{d'n}(Ff) \\ &= \mu_d(Ff) \end{aligned}$$

(the process of composing with Ff preserves sups of chains). ■

(If you're wondering whether there's some underlying principle behind proofs like this, see section 14.6.)

Proposition 6.5.3 *$\mathcal{LSig}$ and $\mathcal{CSig}$, thought of as $\mathcal{Ord}$ -categories, have all (small) weighted limits.*

The reason for this is that $\mathcal{LSig}$ is reflective in $\mathcal{Ord}$ —see 10.5—so you can ‘reflect the weight into $\mathcal{LSig}$ ’ and then the limit is the same as the object constructed above.

Presumably there's then no obstacle to constructing lax limits via weighted limits in the usual way. But I haven't tried to write out the details: partly because (as I mentioned before) there are foundational matters that should be worked out first; and also because the direct construction of lax limits is easier anyway.

Chapter 7

Lifting and strictness

7.1 Focal Σ -spaces

Call an object $X \in \mathcal{E}ff$ **focal** if it has a bottom element $\perp$ under the Σ -order; note that if X is a Σ -space then this bottom element must be unique, so we can think of it as a global section $1 \rightarrow X$. (Thus $\perp$ is a *focal point* of X in the sense of Freyd: the only open set containing it is the whole space.) Then in the usual way we have, in both an external and an internal sense,

Proposition 7.1.1 *If X is a complete focal Σ -space then any $f: X \rightarrow X$ has a least fixed point, namely*

$$Yf = \sup_n f_n(\perp).$$

Let $\mathcal{CF}Sig$ be the full subcategory of complete focal Σ -spaces. It's not complete, of course; but if we call a map $f: X \rightarrow Y$ between focal Σ -spaces **strict** if $f(\perp) = \perp$, then clearly

Proposition 7.1.2 *Let $F: \mathbf{D} \rightarrow \mathcal{CF}Sig$ be a diagram of complete focal Σ -spaces in which every map is strict. Then $X = \varprojlim F$ is a complete focal Σ -space, and each projection $\pi_d: X \rightarrow F(d)$ is strict.*

(See section 7.4 for a better explanation.)

In particular, $\mathcal{CF}Sig$ is closed under indexed products and (hence) is an exponential ideal. It is also closed under retracts.

Proposition 7.1.3 *If a Σ -space X is focal, then so is its chain completion X_G .*

Proof. Let $\perp_C$ be the image of the bottom element $\perp \in X$ under $X \rightarrow X_C$. To show that $\perp_C$ is a (the) bottom element of X_C , we must show that for all $x' \in X_C$, $\perp_C \leq x'$.

That is, we must show that given $U \subseteq_\Sigma X_C$, if $\perp_C \in U$ then $x' \in U$ for all $x' \in X_C$.

But recall that since Σ is a complete Σ -space, pulling back along $X \rightarrow X_C$ induces an isomorphism $\Sigma^X \cong \Sigma^{X_C}$. Let $V \subseteq_\Sigma X$ be the Σ -subset of X corresponding to $U \subseteq_\Sigma X_C$.

Since $\perp_C \in U$, $\perp \in V$. But given any $x \in X$, $x \leq \perp$ and so $x \in V$; that is, $V = X$. Hence $U = X_C$, because both subsets extend $X \subseteq_\Sigma X$ to a Σ -subset of X_C , and such an extension must be unique. ■

Notice that this result is not completely trivial. This is another reminder that working with complete Σ -spaces is quite different from working with ordinary chain-complete posets: often, what is difficult in one setting is easy in the other, and vice-versa.

Remark. Observe that in $\mathcal{CFSig}$ we can solve domain equations by the usual inverse limit method (as in [62]). See 6.2.

7.1.1 On the fixed point category of $\mathcal{E}ff$

Say that an object $X \in \mathcal{E}ff$ is in the **fixed point category** of $\mathcal{E}ff$ if, for any $Y \in \mathcal{E}ff$, every map $X^Y \rightarrow X^Y$ has a fixed point $1 \rightarrow X^Y$ (see also section 14.1). Then by the results above, any complete focal Σ -space, and hence any retract of a power of Σ , is in the fixed-point category. Now, Rosolini proved some time ago, by direct calculation, that any object of the form Σ^X was in the fixed-point category; and it was conjectured at that time that the objects in the fixed-point category were exactly the retracts of the Σ^X . We now show that this is false, using the ideas of [50] (see 2.8).

Let A be some nonrecursive set and consider $\mathcal{E}ff_A$, the realizability topos for A -recursive functions. Then by relativizing the proofs in this chapter, we see that $\mathcal{E}ff_A$ contains an ‘ A -recursive subobject classifier’ $\Sigma_{(A)}$; and that any power of $\Sigma_{(A)}$ is in the fixed-point category of $\mathcal{E}ff_A$.

But $\mathcal{E}ff_A$ is a sheaf subtopos of $\mathcal{E}ff$ (in fact, it’s the sheaves for the least topology making $2 \rightarrow Z_A$ dense, but never mind). And given $X \in \mathcal{E}ff$, $\Sigma_{(A)}^X$ is in $\mathcal{E}ff_A$; so, since $\mathcal{E}ff_A$ is an exponential ideal in $\mathcal{E}ff$, $\Sigma_{(A)}^X$ is in the fixed-point category of $\mathcal{E}ff$.

But unless $X = 0$, $\Sigma_{(A)}^X$ cannot be a Σ -space (for example, $2_A \mapsto \Sigma_{(A)} \mapsto \Sigma_{(A)}^X$ so $\Sigma_{(A)}^X \notin \text{Orth}(2_A)$; see the discussion at the end of section 5.2).

7.2 The lift construction

Given a complete Σ -space X , how can we obtain a complete focal Σ -space? What we would like to do is to 'add a bottom element to X ' to get the lifted domain X_1 . It should have the property that maps $Y \rightarrow X_1$ correspond to computable partial maps $Y \rightarrow X$; that is, X_1 should classify an appropriate class of partial maps with codomain X . But $\tilde{X}$ (the ordinary partial map classifier) and $X + 1$ (which classifies partial maps with decidable domain) are both unsuitable: $\tilde{X}$ classifies too many partial maps, $X + 1$ too few, and neither is a complete focal Σ -space anyway.

We need to decide what a 'computable partial map' $f: Y \rightarrow X$ should be. Writing $Y \cong \mathbb{N}/S, X \cong \mathbb{N}/R$, we would expect f to be tracked by a partial recursive function ψ : that is,

$$\begin{aligned} nSn' &\rightarrow (\psi(n) \downarrow \leftrightarrow \psi(n') \downarrow), \\ nSn' \wedge \psi(n) \downarrow &\rightarrow \psi(n)R\psi(n'). \end{aligned}$$

Intuitively, ψ takes a code for $y \in Y$ to a code for $f(y) \in X$, whenever $f(y)$ is defined. The first condition says that if ψ is defined for some $n \in |y \in Y|$, then it's defined for every $n' \in |y \in Y|$; the second says that ψ respects the partial equivalence relations R and S . It implies that

$$n \in \text{dom } S \rightarrow \psi(n) \in (\text{dom}) R,$$

and we will assume that if ψ is defined for some code for an element $y \in Y$, then $f(y)$ is defined too—that is, we are given a method for computing f such that f is defined whenever the method produces a result. (This is quite a constructive notion of partial map.)

But now $\text{dom } \psi$ defines a Σ -subset of Y ; and conversely, if $Y' \subset_\Sigma Y$ then any $f: Y' \rightarrow X$ is tracked by a partial recursive ψ satisfying the above conditions.—Summing up, we will call a partial map $f: Y \rightarrow X$ **computable** if the domain of f is a Σ -subset of X .

So we would like X_1 to be the r.e. partial map classifier for X ; i.e. X_1 should classify partial maps into X with domains classified by Σ .

$X_\perp$ is constructed in the same way as $\tilde{X}$: it's the product of $\left(\begin{smallmatrix} X \\ 1 \end{smallmatrix}\right)$ along $\top: 1 \rightarrow \Sigma$; that is, $X_\perp = \coprod_\Sigma \prod_\top \left(\begin{smallmatrix} X \\ 1 \end{smallmatrix}\right)$. We call $X_\perp$ the **lift** of X .

It's simple to check that $X_\perp$ has the correct universal property, and that

$$\begin{array}{ccc} X & \rightrightarrows & X_\perp \\ \downarrow & & \downarrow \\ 1 & \xrightarrow{\top} & \Sigma \end{array}$$

is a pullback. In particular, $X \subset_\Sigma X_\perp$. We call its classifying map $\chi_X: X_\perp \rightarrow \Sigma$ the **convergence predicate**; if $t \in X_\perp$ we write $t \Downarrow x$ if $\chi_X(t)$ and t corresponds to the element $x \in X$. This is explained further in the next section.

Note that $1_\perp = \Sigma$; more generally, Sig is closed under the lifted domain construction:

Proposition 7.2.1 *If X is a Σ -space, then so is $X_\perp$.*

Proof. Let $z, z': Z \rightarrow X_\perp$ be two elements of $X_\perp$ satisfying

$$\forall f \in \Sigma^{X_\perp}. fz = fz'$$

Now z and z' classify computable partial maps $\tilde{z}, \tilde{z}': Z \rightarrow X$; it suffices to show that $\tilde{z} = \tilde{z}'$.

Firstly, if $m: X_\perp \rightarrow \Sigma$ classifies $j: X \rightarrow X_\perp$ then $mz: Z \rightarrow X_\perp \rightarrow \Sigma$ classifies $\text{dom } \tilde{z}$. But $mz = mz'$ so $\text{dom } \tilde{z} = \text{dom } \tilde{z}' = Z_0$, say. Let $i: Z_0 \rightarrow Z$ be the inclusion.

Pulling back $z, z': Z \rightarrow X_\perp$ along $j: X \rightarrow X_\perp$ gives maps $z_0, z_0': Z_0 \rightarrow X$; we have to show that $z_0 = z_0'$. Since X is a Σ -space, we only have to establish

$$\forall g \in \Sigma^X. gz_0 = gz_0'$$

Suppose $g \in \Sigma^X$. Since $X \subset_\Sigma X_\perp$, by 5.1.1 we have

$$\exists g_\perp \in \Sigma^{X_\perp}. g = g_\perp j$$

But then

$$gz_0 = g_\perp j z_0 = g_\perp z i = g_\perp z' i = g_\perp j z_0' = gz_0' \quad .$$

■

$X_{\perp}$ has a bottom element under the Σ -order: it's the map $1 \rightarrow X_{\perp}$ which classifies the partial map

$$\begin{array}{ccc} 0 & \xrightarrow{\quad} & 1 \\ \downarrow & & \\ X & & \end{array}$$

We call this bottom element $\emptyset \in X_{\perp}$.

In [68, 3.1], there is the more familiar-looking definition

$$X_{\perp} = \{t \subset X \mid (t \text{ has at most one element}) \wedge (\exists x. x \in t \mid x \in \Sigma)\}.$$

That is, we have a pullback square

$$\begin{array}{ccc} X_{\perp} & \longrightarrow & \tilde{X} \\ \downarrow & & \downarrow \\ \Sigma & \longrightarrow & \Omega \end{array}$$

But here's a warning: the Σ -order on $X_{\perp}$ is *not* in general the inclusion ordering (the order on $X_{\perp}$ thought of as a subposet of the powerset of X). Rather, since $X \subset_{\Sigma} X_{\perp}$, X is a subspace of $X_{\perp}$.

We need to show that lifting behaves well with respect to linked Σ -spaces and complete Σ -spaces (actually, we only need the result for complete Σ -spaces).

Proposition 7.2.2 (i) *If X is a linked Σ -space, then so is $X_{\perp}$.*

(ii) *If X is a complete Σ -space, then $X_{\perp}$ is a complete focal Σ -space.*

Proof. (i) (adapted from Hyland) Let $O_{X_{\perp}}$ be the Σ -order relation on $X_{\perp}$. We want a map $O_{X_{\perp}} \rightarrow X_{\perp}^{\Sigma}$ inverse to $X_{\perp}^{\Sigma} \rightarrow O_{X_{\perp}}$; by the exponential adjunction it suffices to find an appropriate map $O_{X_{\perp}} \times \Sigma \rightarrow X_{\perp}$. We will construct the corresponding computable partial map $O_{X_{\perp}} \times \Sigma \rightarrow X$ by patching together two other computable partial maps with smaller domains.

Firstly, consider the map $O_X \times \Sigma \rightarrow X$ that exists because X is linked. Since $X \subset_{\Sigma} X_{\perp}$, $O_X \subset_{\Sigma} O_{X_{\perp}}$; so $O_X \times \Sigma \subset_{\Sigma} O_{X_{\perp}} \times \Sigma$ and we have a computable partial map $O_{X_{\perp}} \times \Sigma \rightarrow X_{\perp}$.

Secondly, consider the subobject $Y \hookrightarrow O_{X_{\perp}} \times \Sigma$ defined by

$$Y = \{(t, t', p) \mid t \leq t', t' \in X, p = \top\}.$$

$Y \subset_{\Sigma} O_{X_{\perp}} \times \Sigma$ since ' $t' \in X$ ' and ' $p = \top$ ' are Σ -predicates; so the projection $Y \rightarrow X$ defined by

$$(t, t', p) \mapsto t'$$

gives us a computable partial map $O_{X_\perp} \times \Sigma \rightarrow X_\perp$.

It's easy to check that these two partial maps agree where their domains intersect; and since the union of two Σ -subsets is another Σ -subset, patching them together yields a computable partial map. This map is the one we want.

(ii) First note that given a Σ -predicate $p: Y \times \mathbb{N} \rightarrow \Sigma$, there is a computable partial function $\mu n.p: Y \rightarrow \mathbb{N}$. In particular, if $C_{X_\perp} \mapsto X_\perp^\mathbb{N}$ is the object of chains in $X_\perp$ then we have $C_{X_\perp} \rightarrow \mathbb{N}$ taking a chain $f \in X_\perp^\mathbb{N}$ to $\mu n.f(n) \in X$ (the domain of this map is the subobject of 'eventually inhabited chains').

So there is a computable partial map $C_{X_\perp} \rightarrow C_X$ taking

$$f \mapsto [m \mapsto f(m + \mu n.f(n) \in X)].$$

But X is a complete Σ -space so we have a map $C_X \rightarrow X$ taking a chain in X to its sup; hence we have a computable partial map $C_{X_\perp} \rightarrow X$ taking eventually inhabited chains to their sups (which of course lie in X). But this just corresponds to a map $C_{X_\perp} \rightarrow X_\perp$, which is easily seen to take any chain in $X_\perp$ to its sup in $X_\perp$. ■

We should state a simple fact which will be very useful in section 7.4.

Proposition 7.2.3 $X + 1$ is dense in $X_\perp$ (where the second summand of $X + 1$ corresponds to the bottom element $\emptyset$ of $X_\perp$).

Proof. This simply says that $X \vee \neg X \mapsto X_\perp$; but X is a Σ -subset of $X_\perp$, hence $\neg\neg$ -closed, and so this follows from 2.4.4. ■

Finally, the Σ -subsets of $X_\perp$ have quite a simple description in terms of the Σ -subsets of X . Roughly speaking, $\Sigma^{X_\perp}$ is ' Σ^X together with an extra top element'.

Proposition 7.2.4 Let $X \in \mathcal{CSig}$. Given $p \in \Sigma$ and a predicate $\phi \in \Sigma^X$ such that $\forall x \in X. p \leq \phi(x)$, ϕ extends to a unique predicate $\psi \in \Sigma^{X_\perp}$ with

$$\psi(t) = \begin{cases} \phi(x) & t \downarrow = x \\ p & t \uparrow; \end{cases}$$

furthermore, all predicates $\psi \in \Sigma^{X_\perp}$ arise in this way.

See also [30].

Proof. (Sketch.) The proof follows the same lines as the proofs in section 7.4. To construct ψ , approximate $|t \in X|$ by decidable truth-values and use definition-by-cases to construct a suitable chain whose sup is $\psi(t)$; to show that any $\psi \in \Sigma^{X_\perp}$ is uniquely determined by the corresponding ϕ and p , note that ψ is determined by its values on the $\neg\neg$ -dense subobject $X + 1 \rightarrow X_\perp$. ■

7.3 Computational interpretation of lifting

In this section we give an explicit description of the lifted domain $X_\perp$ as a partial equivalence relation on $\mathbf{N}$. It is almost identical to the one discovered independently by the authors of [21]. We also reprove 7.2.2 using recursion-theoretic methods; these proofs (which were the original ones) are actually the ‘same’ as the ones before.

Again everything is written out informally; it would be easy to translate everything into the internal language, using T -predicates and so on, but that would make things unreadable.

Proposition 7.3.1 *Let $X \cong \mathbf{N}/R$ be a modest set. Then $X_\perp \cong \mathbf{N}/R_\perp$, where the partial equivalence relation $R_\perp$ is defined by*

$$\begin{aligned} eR_\perp e & \text{ iff } (W_e \text{ has at most one element}) \\ & \quad \wedge (n \in W_e \rightarrow nRn); \\ eR_\perp e' & \text{ iff } eR_\perp e \wedge e'R_\perp e' \wedge \\ & \quad (W_e \text{ is inhabited iff } W_{e'} \text{ is inhabited}) \\ & \quad \wedge (n \in W_e \wedge n' \in W_{e'} \rightarrow nRn'). \end{aligned}$$

[Note: This shows that $X_\perp$ is the set of ‘partial computations of elements of X ’, i.e. computations that, if they terminate, yield a single code for an element of X . You should check that the description of $R_\perp$, when written out formally in the internal language, really defines a $\neg\neg$ -closed partial equivalence relation on $\mathbf{N}$.]

Proof. It suffices to show that, if $Y \cong \mathbf{N}/S$ is a modest set, then there is a natural bijection between maps $Y \rightarrow X_\perp$ and partial maps

$$\begin{array}{c} Y' \subseteq_\Sigma Y \\ \downarrow \\ X \end{array}$$

(Why? Well, $\mathcal{Mod}$ is a reflective subcategory of $\mathcal{Eff}$; so since X and $X_\perp$ are modest, if Y is an arbitrary object then any map $Y \rightarrow X_\perp$ factors through the modest reflection of Y , and any partial map

$$\begin{array}{c} Y' \subset_\Sigma Y \\ \downarrow \\ X \end{array}$$

factors through the modest reflection of Y' , which preserves the property of being a Σ -subset.) We will sketch how to go from one to the other, and leave the reader to check that this yields a bijection which is natural.

Suppose $Y' \subset_\Sigma Y$ and $f: Y' \rightarrow X$ is tracked by ψ . Then there is a map $\hat{f}: Y \rightarrow X_\perp$ tracked by the partial recursive function $\hat{\psi}$, where

$$W_{\hat{\psi}(n)} = \begin{cases} \{\psi(n)\} & \text{if } \psi(n) \downarrow \\ \phi & \text{if } \psi(n) \uparrow. \end{cases}$$

Conversely, given $\hat{f}: Y \rightarrow X_\perp$ tracked by the partial recursive function $\hat{\psi}$, define a partial recursive function ψ as follows: to compute $\psi(n)$, (attempt to) list an element of $W_{\hat{\psi}(n)}$. Then $\text{dom } \psi$ defines a Σ -subset $Y' \subset_\Sigma Y$, and ψ tracks a map $f: Y' \rightarrow X$. ■

Corollary 7.3.2 *Let $X \cong \mathbb{N}/R$, $Y \cong \mathbb{N}/S$ be modest. Then any map $f: X \rightarrow Y_\perp$ extends uniquely to a strict map $f_\perp: X_\perp \rightarrow Y_\perp$.*

Proof. The existence of $f_\perp$ is a special case of [68, proposition 3.2.4(iv)], since $X \subset_\Sigma X_\perp$; but here's a different proof. We leave the reader to check uniqueness.

Suppose ψ tracks f , and define $\psi_\perp$ as follows:

$$W_{\psi_\perp(e)} = \begin{cases} W_{\psi(n)} & \text{if } n \in W_e \text{ and } \psi(n) \downarrow \\ \phi & \text{otherwise.} \end{cases}$$
■

(See the next section for a more synthetic proof.)

Proposition 7.3.3 (i) *If X is a linked Σ -space, then so is $X_\perp$.*
(ii) *If X is a complete Σ -space, then $X_\perp$ is a complete focal Σ -space.*

Proof. (i) Let $X \cong \mathbb{N}/R$ and let $X_\perp \cong \mathbb{N}/R_\perp$ as described above. Since X is linked, we have a map $O_X \rightarrow X^\Sigma$ taking each $x \leq x'$ to a map that links them; let ψ be a partial recursive function tracking this map. This means that if m, m' are codes for x, x' then $\psi(m, m')$ is an index for (a total recursive function that tracks) a map $\text{Sig} \rightarrow X$ linking x and x' ; i.e.

$$\begin{aligned} & \{\psi(m, m')\}(n)Rm \quad \text{if } n \notin K \\ & \{\psi(m, m')\}(n)Rm' \quad \text{if } n \in K. \end{aligned}$$

Now, let $e, e' \in \text{dom } R_\perp$ be codes for $t, t' \in X_\perp$ where $t \leq t'$. This means that

$$\begin{aligned} & W_e \neq \phi \rightarrow W_{e'} \neq \phi \\ & (m \in W_e \wedge m' \in W_{e'}) \rightarrow [m] \leq [m'] \end{aligned}$$

(where $[m]$ denotes the equivalence class of m). Then from e and e' we can effectively compute an index $\bar{\psi}(e, e')$ for (a recursive function tracking) a map $\Sigma \rightarrow X_\perp$ linking t and t' as follows: Let $\bar{\psi}(e, e')$ be an index for a recursive function ξ satisfying the following requirement:

$W_{\xi(n)}$ is the set consisting of the result of the following computation, if it terminates; otherwise $W_{\xi(n)} = \phi$.

Step 1: List $W_{e'}$ till an element m' appears (if this happens, then $W_{e'} = \{m'\}$).

Step 2: Simultaneously carry out the following computations, till one of them produces a result; then output that result:

- (a) compute $\{n\}(n)$, then output m' .
- (b) list W_e till an element m appears, then output $\{\psi(m, m')\}(n)$.

Now, if step 1 doesn't terminate then $W_{e'} = \phi$ and $W_{\xi(n)} = \phi$. And if step 1 terminates and $n \notin K$ then (a) doesn't terminate but either (b) terminates yielding an element of $[m]$, or $W_e = \phi$ and (b) doesn't terminate so $W_{\xi(n)} = \phi$ again. And if step 1 terminates and $n \in K$ then (a) and (b) will both terminate and yield an element of $[m']$ (so it doesn't matter which terminates first). Hence

$$\begin{aligned} & \xi(n)R_\perp e \quad \text{if } n \notin K \\ & \xi(n)R_\perp e' \quad \text{if } n \in K; \end{aligned}$$

so ξ tracks a map linking t and t' , as required.

(ii) Since X is complete, there is a map $C_X \rightarrow X$ taking a chain (x_n) to $\sup_n x_n$; let ψ be a partial recursive function tracking this map. This means that if e is an index such that

$$\begin{aligned} \{e\}(n) &\in \text{dom } R && \text{for all } n, \text{ and} \\ [\{e\}(n)] &\leq [\{e\}(n+1)] && \text{in } X \cong \mathbb{N}/R, \text{ for all } n \end{aligned}$$

then $[\psi(e)] = \sup_n [\{e\}(n)]$ in $\mathbb{N}/R$.

Now, suppose we have a chain (t_n) in $X_\perp$; this means we have an index f such that

$$\begin{aligned} \{f\}(n) &\in \text{dom } R_\perp, && \text{for all } n \\ W_{\{f\}(n)} &\neq \phi \rightarrow W_{\{f\}(n+1)} \neq \phi \\ (m \in W_{\{f\}(n)} \wedge m' \in W_{\{f\}(n+1)}) &\rightarrow [m] \leq [m']. \end{aligned}$$

Then from f we can effectively compute a number $\bar{\psi}(f)$ such that

$$[\bar{\psi}(f)] = \sup_n [\{f\}(n)] \quad \text{in } \mathbb{N}/R$$

as follows:

Let $W_{\bar{\psi}(f)}$ be the set consisting of the result of the following computation, if it terminates; otherwise $W_{\bar{\psi}(f)} = \phi$:

Step 1: Simultaneously enumerate all the $W_{\{f\}(n)}$ till you find an N such that $W_{\{f\}(N)} \neq \phi$.

Step 2: Let e be an index such that $\{e\}(n)$ is the sole member of $W_{\{f\}(N+n)}$; now output $\psi(e)$.

■

If we compare these proofs to the previous ones, we see how taking the synthetic point of view, and exploiting the internal logic of $\mathcal{E}\mathcal{F}$, makes things much simpler for us. Also, observe that as presented here, the proofs are apparently valid only in the fibre over 1; it takes some care to verify that they can be internalized, so as to be valid in any slice of $\mathcal{E}\mathcal{F}$.

7.4 Strictness and the lift monad

We have seen that the lift construction is the object part of a functor $(-)_\perp : \mathcal{C}\mathcal{S}\mathcal{ig} \rightarrow \mathcal{C}\mathcal{S}\mathcal{ig}$, whose image actually lies in the category $\mathcal{C}\mathcal{F}\mathcal{S}\mathcal{ig}_\perp$ of complete focal Σ -spaces

and strict maps. In this section we present a characterization of $\mathcal{CFSig}_\perp$ first noticed by Hyland: $(-)_\perp$ is the functor part of a monad on $\mathcal{CSig}$, whose category of algebras is equivalent to $\mathcal{CFSig}_\perp$.

Recall from [68] that an object $Y \in \mathcal{Eff}$ is called Σ -injective if, whenever we have an object X , a Σ -subset $U \subset_\Sigma X$ and a map $f: U \rightarrow Y$, f extends to a map $\bar{f}: X \rightarrow Y$.

Lemma 7.4.1 *Any complete focal Σ -space Y is Σ -injective.*

Proof. Given $x \in X$ there is a chain (p_n) of decidable truth-values such that $x \in U \leftrightarrow \exists n. p_n$. Define a chain (y_n) in Y by

$$y_n = \begin{cases} f(x) & p_n; \\ \perp & \neg p_n \end{cases}$$

and let $\bar{f}(x) = \sup_n y_n$.

To see that this map is well-defined, let $(p_n), (p'_n)$ be two chains of decidable truth-values such that

$$x \in U \leftrightarrow \exists n. p_n \leftrightarrow \exists n. p'_n,$$

let $(y_n), (y'_n)$ be the corresponding chains in Y as defined above, and let $y = \sup_n y_n$ and $y' = \sup_n y'_n$. By considering the cases $p_n = \perp, p_n = \top$ we see

$$\forall n. \exists m. p_n \leq p'_m;$$

so (y'_n) is cofinal in (y_n) and thus $y \leq y'$. Similarly $y' \leq y$ and so $y = y'$. ■

Proposition 7.4.2 *Let Y be a complete focal Σ -space, X a complete Σ -space. Then any map $f: X \rightarrow Y$ extends uniquely to a strict map $f_\perp: X_\perp \rightarrow Y$.*

Proof. Since $X \subset_\Sigma X_\perp$, we can apply the lemma to get an extension $f_\perp$ of f . It is easy to check that the map $f_\perp$ is strict.

As for uniqueness, observe that any element $t \in X_\perp$ is the supremum of a chain (t_n) where each t_n is (decidably) either $\perp$ or a member of X . The action of f on t_n is determined; so by continuity, the action of f on t is determined.

(Alternatively, observe that $f_\perp$ is determined by its restriction to the $\neg\neg$ -dense subobject $X + 1 \hookrightarrow X_\perp$.) ■

We can use proposition 7.4.2 to describe the monad structure on $(-)_\perp$. The unit of the monad is the inclusion $i_X: X \mapsto X_\perp$; and one uses the proposition to construct the multiplication for the monad in the obvious way. The associative and unitary identities are proved in the same way as the uniqueness statement in 7.4.2.

Corollary 7.4.3 *Let X be a complete focal Σ -space. Then there is a unique map $r_X: X_\perp \rightarrow X$ giving X the structure of an algebra for the lift monad on $\mathcal{CSig}$.*

Proof. Let r_X be the unique strict extension of the identity map $X \rightarrow X$ to $X_\perp$; clearly $r_X i_X = 1$.

r_X is the unique such map. To see this, we must show that if $r': X_\perp \rightarrow X$ satisfies $r' i_X = 1$ then $r' = r_X$. It suffices to show that the two maps agree on $X + 1 \mapsto X_\perp$; so in fact, all we must show is that $r'(\emptyset) = \perp$. But for all $x \in X$, $\emptyset \leq x$ and so $r'(\emptyset) \leq r' i_x(x) = x$.

It remains to check that

$$\begin{array}{ccc} (X_\perp)_\perp & \xrightarrow{r_{X_\perp}} & X_\perp \\ (r_X)_\perp & \downarrow & \downarrow r_X \\ X_\perp & \xrightarrow{r_X} & X \end{array}$$

But we can easily check that both ways round agree on the $\neg\neg$ -dense subobject $(X + 1) + 1 \mapsto (X_\perp)_\perp$. ■

If X and Y are complete focal Σ -spaces then one can verify that a map $f: X \rightarrow Y$ is strict if and only if it's an algebra map. So

Corollary 7.4.4 *The category of algebras for the lift monad on $\mathcal{CSig}$ is equivalent to $\mathcal{CFSig}_\perp$.*

Proof. Suppose X and Y are complete focal Σ -spaces; we have to verify that a map $f: X \rightarrow Y$ is strict if and only if it's an algebra map. So suppose f is strict; then the fact that

$$\begin{array}{ccc} X_\perp & \xrightarrow{f_\perp} & Y_\perp \\ r_X \downarrow & & \downarrow r_Y \\ X & \xrightarrow{f} & Y \end{array}$$

is easily checked on $X + 1 \mapsto X$. Conversely if f is an algebra map then

$$f(\perp_X) = f r_X(\emptyset_X) = r_Y f_\perp(\emptyset_X) = r_Y(\emptyset_Y) = \perp_Y$$

(where $\emptyset_X$ and $\emptyset_Y$ are the bottom elements of $X_\perp$ and $Y_\perp$). ■

Proposition 7.4.5 *The category of free algebras for the lift monad (the Kleisli category) is equivalent to the category of complete Σ -spaces and computable partial maps.*

Proof. Proposition 7.4.2 implies that strict maps $X_\perp \rightarrow Y_\perp$ are in bijection with maps $X \rightarrow Y_\perp$. ■

The following result says that the free algebras are exactly the ‘complete Σ -spaces with closed generic point’ as studied in [74]:

Proposition 7.4.6 *Let X be a complete focal Σ -space, and let*

$$D = \{x \in X \mid x \neq \perp\}.$$

Then X is a free algebra if and only if $D \subseteq_\Sigma X$.

Proof. If $X = Y_\perp$ then clearly $D = Y \subseteq_\Sigma Y_\perp = X$. Conversely if $D \subseteq_\Sigma X$, we will show that $X = D_\perp$; i.e. that for any object Z , maps $Z \rightarrow X$ correspond to computable partial maps $Z \rightarrow D$.

Given $Z \rightarrow X$, pulling back along $D \hookrightarrow X$ gives us a partial map $Z \rightarrow D$ with domain a Σ -subset. On the other hand, suppose we have a computable partial map

$$\begin{array}{ccc} Z' & \subseteq_\Sigma & Z \\ \downarrow & & \\ D & & \end{array}.$$

Now the proof goes the same way as in 7.4.1. Let $\phi: Z \rightarrow \Sigma$ classify Z' ; then for each $z \in Z$, there exists a chain (p_n) of decidable truth-values such that

$$\phi(z) \leftrightarrow \exists n. p_n.$$

Define a chain (x_n) in X by

$$x_n = \begin{cases} f(z) & p_n \quad (\text{so } z \in Z') \\ \perp & \neg p_n \end{cases}$$

and let $f(z) = \sup_n x_n$; this gives us the desired map from Z to X . ■

Since $\mathcal{CSig}$ is complete, $\mathcal{CFSig}_\perp$ is also complete. So by the adjoint functor theorem there exist functors

$$\begin{aligned}\oplus: \mathcal{CFSig}_\perp^2 &\rightarrow \mathcal{CFSig}_\perp && \text{---'coalesced sum';} \\ \otimes: \mathcal{CFSig}_\perp^2 &\rightarrow \mathcal{CFSig}_\perp && \text{---'smash product'.}\end{aligned}$$

$\oplus$ is left adjoint to the diagonal functor and $\otimes$ is left adjoint to the strict function space functor. In [74], Scott gave an alternative construction which works only for free algebras.

$\mathcal{CFSig}_\perp$ is small and complete, and hence has all coequalizers. But it's often useful to know how to calculate coequalizers in categories of algebras. The proof of the following fact begins by describing coequalizers as limits of a certain kind.

Proposition 7.4.7 *The forgetful functor $\mathcal{CFSig}_\perp \rightarrow \mathcal{CSig}$ reflects coequalizers.*

Proof. Let X, Y be complete focal Σ -spaces, $f, g: X \rightarrow Y$ strict maps. To show that the coequalizers of f and g in $\mathcal{CFSig}_\perp$ and $\mathcal{CSig}$ are the same, we use the fact that colimits can be described as limits, and that $\mathcal{CFSig}_\perp \rightarrow \mathcal{CSig}$ reflects limits.

Let $Y \rightarrow Q$ be the coequalizer of f and g in $\mathcal{CSig}$, then $Y \rightarrow Q$ is the initial object of the category $\mathbf{C}$ whose objects are maps $Y \rightarrow Z$ in $\mathcal{CSig}$ coequalizing f and g , and whose arrows are commutative triangles. Similarly, the coequalizer of f and g in $\mathcal{CFSig}_\perp$ is the initial object of the subcategory $\mathbf{C}_\perp$ of $\mathbf{C}$ in which the Z are focal and all the maps involved are strict.

Now the initial object of a category is the limit of the identity functor, thought of as a diagram in that category; so to show that the initial objects of $\mathbf{C}$ and $\mathbf{C}_\perp$ are the same, it suffices to prove that $\mathbf{C}_\perp$ is initial in $\mathbf{C}$. We must show that if $h: Y \rightarrow Z$ is a map in $\mathcal{CSig}$ coequalizing f and g , then h has a best possible factorization

$$\begin{array}{ccc} Y & \xrightarrow{h'} & Z' \\ & \searrow & \nearrow \\ & Z & \end{array}$$

where Z' is a complete focal Σ -space and h' is strict and coequalizes f and g .

But $Z'' = \text{Im}(h)$ is focal—its bottom element is the image of the bottom element of Y . So we can let Z' be the chain completion of Z'' , which is a complete focal Σ -space by 7.1.3. ■

That is, $\mathcal{CFSig}_\perp$ has the coequalizers we would expect it to have.

7.5 Projection analysis

Why do we care whether a complete focal Σ -space X is a free algebra (i.e. the lift of something)? Usually it's because we want to make use of 7.4.6; that is, we exploit the fact that $D \subset_\Sigma X$ where

$$D = \{x \in X \mid x \neq \perp\}.$$

The point is that, as in the proof of 7.4.1, we can then define a function with domain X by saying how it acts on D and on $\perp_X$.

One example is in the technique of *projection analysis* for functional programs: see [26]. This is not the right place to give an exposition; so we'll just see that it's possible to carry out certain constructions on which projection analysis depends. We are simply looking (again) at what is and isn't lost in the move from classical domain theory to this new framework.

Let $X \in \mathcal{CFSig}$. Call $\alpha: X \rightarrow X$ a **projection** if $\alpha^2 = 1$ and $\alpha \leq 1$. (I apologise for the clash of terminology with 8.4.) We think of α as 'throwing away' information about $x \in X$ that we're not interested in—so analysing different (usually strictness-related) properties of a program corresponds to singling out different projections α . Certain special kinds of projections turn out to be particularly useful.

Call α a **smash projection** if for all $x \in X$, $\alpha(x) \neq \perp \rightarrow \alpha(x) = x$.

Proposition 7.5.1 *Let X be a free algebra. Then if $\alpha: X \rightarrow X$ is any projection, there is a least smash projection α' greater than α .*

Proof. As in 7.4.1, let $\phi: X \rightarrow \Sigma$ classify D . If $x \in X$, let (p_n) be a chain of decidable truth-values with supremum $\phi(\alpha(x))$, and define

$$x'_n = \begin{cases} x & p_n \\ \perp & \neg p_n. \end{cases}$$

Now let $\alpha'(x) = \sup_n x'_n$; then

$$\alpha'(x) = \begin{cases} x & \alpha(x) \neq \perp \\ \perp & \alpha(x) = \perp \end{cases}$$

■

Call a projection $\alpha: X_\perp \rightarrow X_\perp$ **lift-strict** if $\alpha(\perp) = \emptyset$ (where $\perp$ is the bottom element of X and $\emptyset$ is the bottom element of $X_\perp$.)

Proposition 7.5.2 *Suppose X is a free algebra. Then there is a greatest lift-strict projection $\alpha: X_{\perp} \rightarrow X_{\perp}$.*

Proof. By the same method as before, we can show that there is a unique map $\alpha: X_{\perp} \rightarrow X_{\perp}$ such that

$$\alpha(t) = \begin{cases} \emptyset & t \uparrow \\ \emptyset & t \downarrow = \perp \\ t & t \downarrow = x, x \neq \perp \end{cases}$$

This is clearly the map we want. ■

10.2.4 gives a sufficient condition for a partial function space to be a free algebra; it applies to a large number of the cases we're interested in.

Chapter 8

Coproducts and fibres

8.1 Preliminary remarks

Our goal in this chapter is to prove some theorems about indexed coproducts of Σ -spaces. For this, we need to use the language of fibred categories.

For practical purposes we really only need binary coproducts; but it's of interest to prove some much more general results about internal coproducts. General topos theory tells us that $\mathcal{E}ff \times \mathcal{E}ff \cong \mathcal{E}ff/2$, so binary coproducts are the same as 2-indexed coproducts (internal coproducts along $2 \rightarrow 1$).

Note. In this chapter we'll be a bit casual and talk about products/coproducts only along $\alpha: J \rightarrow I$ of the form $I \rightarrow 1$; the proofs for general α are the same but messier to write.

Another note. We'll often write $(X_i)_{i \in I}$ for an I -indexed family $(\begin{smallmatrix} X \\ I \end{smallmatrix}) \in \mathcal{E}ff/I$; intuitively, $X_i = p^{-1}(i)$. The unit $(\begin{smallmatrix} X \\ I \end{smallmatrix}) \mapsto I^*X = (\begin{smallmatrix} X \times I \\ I \end{smallmatrix})$ (which comes from *graph* $p \mapsto X \times I$) can be thought of as the I -indexed family of inclusions $(m_i: X_i \mapsto X)_{i \in I}$.

Before discussing Σ -spaces, we first explain why the situation for modest sets is much simpler. Recall from [33] that $\mathcal{M}od$ is equivalent to the category $\mathcal{S}ep \cap \mathcal{O}rth(\nabla 2)$ of $\neg\neg$ -separated objects orthogonal to the $\neg\neg$ -closed subobject classifier $\nabla 2$; and note that if $I \in \mathcal{S}ep$ and $(\begin{smallmatrix} X \\ I \end{smallmatrix}) \in \mathcal{S}ep$ then $X \in \mathcal{S}ep$.

Proposition 8.1.1 *Let A be any object in a topos. If I is orthogonal to A and $(\begin{smallmatrix} X \\ I \end{smallmatrix}) = (X_i)_{i \in I}$ is an I -indexed family of objects all orthogonal to A , then X is orthogonal to A .*

Proof. Suppose $f \in X^A$; we must show that f is a constant function. But since I is orthogonal to A , $pf \in I^A$ is constant; say $pf \equiv i$. Then f factors through X_i , i.e. $f = m_i f_i$ for some $f_i \in X_i^A$. But X_i is orthogonal to A , and so f_i is a constant function. ■

It follows that if we have a family of modest sets indexed by a modest set, their coproduct is again a modest set. A similar statement also holds for the category $\mathcal{L}$ defined at the end of 5.2.

8.2 Coproducts

For *Sig*, the situation is not so good. Thomas Streicher (private communication) has found an I -indexed family $(\overset{X}{\underset{I}{p}})$ of Σ -spaces, where I is a Σ -space, such that X is *not* a Σ -space (cf. [75]). In any case, when X does happen to be a Σ -space the problem of describing the Σ -order on it seems difficult. But if we impose a certain condition on the display map $p: X \rightarrow I$, the situation becomes much simpler. Unfortunately this condition is so restrictive that the results obtained here can't be regarded as very satisfactory.

Call an arbitrary family $(\overset{X}{\underset{I}{p}})$ Σ -fibred if $\text{graph } p \subset_{\Sigma} X \times I$; in the internal logic, this means that for all i , $X_i \subset_{\Sigma} X$.

Proposition 8.2.1 *Let $(\overset{X}{\underset{I}{p}})$ be a Σ -fibred family of Σ -spaces, where I is a Σ -space. Then X is a Σ -space.*

Proof. Suppose $x, x' \in X$ satisfy $\forall f \in \Sigma^X. fx = fx'$; then

$$\forall g \in \Sigma^I. gpx = gpx';$$

so, since I is a Σ -space, $px = px' = i$, say. This means that x and x' lie in X_i , i.e. that $x = m_i x_i$, $x' = m_i x'_i$ for some $x_i, x'_i \in X_i$.

We want to show that $x_i = x'_i$; since X_i is a Σ -space, it suffices to show that $\forall h \in \Sigma^{X_i}. hx_i = hx'_i$. Suppose then that $h \in \Sigma^{X_i}$. Now since $X_i \subset_{\Sigma} X$, by 5.1.1 we have $\exists f \in \Sigma^X. h = fm_i$. But then

$$hx_i = fm_i x_i = fx = fx' = fm_i x'_i = hx'_i$$

which is what we wanted. (Note that we only used the fact that $\Sigma^X \rightarrow \Sigma^{X_i}$ was an epi.) ■

Note that if the equality relation on I is a Σ -subset of $I \times I$ then any I -indexed family $(\frac{X}{I})$ is Σ -fibred. Since this is true for $I = 2$ and $\mathbb{N}$, $\mathbf{Sig}$ is closed under binary (and countable) coproducts. And since all these arguments work in any fibre of $\mathcal{E}ff$, the same is true of $\mathbf{Sig}$ in an internal sense.

The assumption that the family is Σ -fibred means that the Σ -order on the coproduct is the one we would expect.

Proposition 8.2.2 *Let I be a Σ -space and let $(\frac{X}{I})$ be a Σ -fibred family of Σ -spaces; suppose $x, x' \in X$. Then $x \leq x'$ in X if and only if $px = px' = i$ and $x \leq x'$ in X_i .*

Proof. (The 'if' part is trivial since m_i is monotone.) If $x \leq x'$ in X then $px \leq px'$ in I . Since $\text{graph } p \subset_{\Sigma} X \times I$, $I' = \{i' \in I \mid px = i'\} \subset_{\Sigma} I$. But $px \in I'$, so $px' \in I'$; i.e. $px = px' = i$, say; thus x, x' both lie in X_i . But $X_i \subset_{\Sigma} X$, so X_i is a subspace of X ; the result follows. ■

Now we show that if each X_i is a linked/complete/flat Σ -space, then so is $X = \coprod_i X_i$.

Proposition 8.2.3 *Suppose $(\frac{X}{I})$ is a Σ -fibred family of Σ -spaces.*

- (i) *If $(\frac{X}{I}) \in \mathcal{LSig}(I)$ then $X \in \mathcal{LSig}$.*
- (ii) *If $(\frac{X}{I}) \in \mathcal{CSig}(I)$ then $X \in \mathcal{CSig}$.*
- (iii) *If $(\frac{X}{I}) \in \mathcal{FSig}(I)$ then $X \in \mathcal{FSig}$.*

Proof. (i): Suppose $x, x' \in X$ with $x \leq x'$. Then they both lie in some X_i ; that is, $x = m_i x_i, x' = m_i x'_i$ for some $x_i, x'_i \in X_i$. X_i is linked, so let $f \in X_i^{\Sigma}$ be a map linking x_i and x'_i ; then $m_i f \in X^{\Sigma}$ links x and x' .

(ii): Let (x_n) be a chain in X . Since

$$x_0 \leq x_1 \leq x_2 \leq \dots,$$

we have

$$px_0 = px_1 = px_2 = \dots = i,$$

say; that is, the (x_n) all lie in X_i . Again, compute their sup there, and apply m_i (which preserves sups of chains: see 5.5.1).

(iii): Trivial. ■

There are likely to be many interesting families of Σ -spaces that are not Σ -fibred; in fact, the class of Σ -fibred families is not a very nice one (it is not even closed under pullback: the family $(\frac{1}{1})$ is Σ -fibred but $(\frac{\Sigma}{1})$ is not, since equality on Σ is not a Σ -relation). However, a good theory is lacking. Note that one can't expect an analogue of 8.2.2 to hold for sums of more general families.

8.3 Fibres

In chapter V of [76], Taylor describes classes of display maps in classical categories of domains: these are continuous maps $D \rightarrow E$ of domains that can sensibly be thought of as 'displaying' an E -indexed family of domains (the family of fibres of the map). For Σ -spaces, the situation is simple: *any* way of fibring a Σ -space gives a family of Σ -spaces. Furthermore, any 'computable' way of fibring a complete Σ -space gives a family of complete Σ -spaces.

Proposition 8.3.1 *Let $p: X \rightarrow I$ be a map in $\mathcal{E}ff$.*

- (i) *If $X \in \text{Sig}$, $(\frac{X}{I}) \in \text{Sig}(I)$.*
- (ii) *If $X \in \mathcal{L}\text{Sig}$ and $I \in \text{Sep}$, $(\frac{X}{I}) \in \mathcal{L}\text{Sig}(I)$.*
- (iii) *If $X \in \mathcal{C}\text{Sig}$ and $I \in \text{Sig}$, $(\frac{X}{I}) \in \mathcal{C}\text{Sig}(I)$.*
- (iv) *If $X \in \mathcal{F}\text{Sig}$, $(\frac{X}{I}) \in \mathcal{F}\text{Sig}(I)$.*

Proof. (i): Suppose $x_i, x_i' \in X_i$ satisfy

$$\forall f \in \Sigma^{X_i}. f x_i = f x_i'$$

Given any $f \in \Sigma^X$, we have $f m_i \in \Sigma^{X_i}$; so $f m_i x_i = f m_i x_i'$. But X is a Σ -space so $m_i x_i = m_i x_i'$; so, since m_i is mono, $x_i = x_i'$.

(ii): $X_i \rightarrowtail X$ is the equalizer of $p, \tau: X \rightarrow I$ (where τ is the constant function at i); hence it is $\neg\neg$ -closed, and the result follows from 5.4.6.

(iii): Let (x_{in}) be a chain in X_i ; then $(m_i x_{in})$ is a chain in X . Let $x = \sup_n m_i x_{in}$; since p preserves sups of chains (5.5.1),

$$p x = \sup_n p m_i x_{in} = \sup_n i = i$$

and so $x = m_i x_i$ for some $x_i \in X_i$. But (ii) tells us that X_i is a subspace of X ; therefore $x_i = \sup_n x_{in}$.

(iv): Suppose $x_i \leq x_i'$ in X_i ; then $m_i x_i \leq m_i x_i'$ in X . So, since X is flat, $m_i x_i = m_i x_i'$ and hence $x_i = x_i'$. ■

Note that (ii) and (iii) are not valid without some assumptions on I . For example, let Z_A be the non-linked Σ -space described in 5.4.2, and let $p: \Sigma \rightarrow \Omega$ classify $Z_A \mapsto \Sigma$; then Σ is a linked Σ -space but $(\frac{\Sigma}{\Omega})_p$ is not a family of linked Σ -spaces (since the fibre over $\top$ is Z_A). Also, let Fin be the object of $\neg\neg$ -finite r.e. subsets of $\mathbb{N}$, and let $p: \Sigma^{\mathbb{N}} \rightarrow \nabla 2$ classify $Fin \mapsto \Sigma^{\mathbb{N}}$; then $\Sigma^{\mathbb{N}}$ is a complete Σ -space but $(\frac{\Sigma^{\mathbb{N}}}{\nabla 2})_p$ is not a family of complete Σ -spaces (since the fibre over $\top$ is Fin).

$(\frac{X}{I})$ can be a very odd family of Σ -spaces. For example, since $\mathbb{N}$ is a Σ -space, the quotient map $\mathbb{N} \twoheadrightarrow \Sigma$ gives a Σ -indexed family of Σ -spaces; that is, $(\frac{\mathbb{N}}{\Sigma}) \in \text{Sig}(\Sigma)$. Now $\prod_{\Sigma} (\frac{\mathbb{N}}{\Sigma})$ is (internally) the set of sections $\Sigma \rightarrow \mathbb{N}$ of $\mathbb{N} \twoheadrightarrow \Sigma$; but $\mathbb{N}$ is orthogonal to Σ , and so $\prod_{\Sigma} (\frac{\mathbb{N}}{\Sigma}) = 0$. (This is not so surprising if we regard it as a failure of the axiom of choice.)

8.4 Projections

We saw in the last section that any map $p: Y \rightarrow X$ of complete Σ -spaces can be thought of as displaying an X -indexed family $(Y_x)_{x \in X}$ of complete Σ -spaces. That is, in $\mathcal{CSig}$, ‘all maps are display maps’. But when are all the Y_x focal?

Recall that p is a **projection** if there exists a map $i: X \rightarrow Y$ with $pi = 1_X, ip \leq 1_Y$. It turns out that a map is a projection precisely when it displays an X -indexed family of complete *focal* Σ -spaces. This follows from:

Proposition 8.4.1 *Let $Y \in \mathcal{LSig}$ and $X \in \text{Sep}$ and let $p: Y \rightarrow X$, as in 8.3.1 (ii). Then the following are equivalent:*

- $\forall x. (Y_x \text{ is focal}).$
- p is a projection with corresponding injection $i: X \rightarrow Y$.

Proof. Suppose p is a projection; we’ll show that $\perp_{Y_x} = ix$. Let $y \in Y$ be an element of Y_x , i.e. $py = x$; then

$$ix = ipy \leq y \quad \text{in } Y.$$

But Y_x is a subspace of Y (cf. the proof of 8.3.1 (ii)) and so $ix \leq y$ in Y_x .

Conversely, suppose $\forall x. (Y_x \text{ is focal})$; this means that $(\frac{Y}{X}p)$ is a focal Σ -space in $\mathcal{E}ff/X$. That is, we have a global section $(\frac{X}{X}1) \rightarrow (\frac{Y}{X}p) \in \mathcal{E}ff/X$ that 'names' the unique bottom element of each Y_x .

Thus we have a map $i: X \rightarrow Y$ with $pi = 1_X$. And if $y \in Y$ then

$$ipy = \perp_{Y_x} \leq y \quad \text{in } Y_x$$

(where $x = py$); but $Y_x \hookrightarrow Y$ is monotone, so $ipy \leq y$ in Y ; that is, $ip \leq 1_Y$. ■

See [44] for an application of these results to partial evaluation in the case when p is a projection. (Actually, he had to prove the corresponding result for Scott domains, which is much more complicated.)

Chapter 9

Algebraic constructions

9.1 The convex powerdomain

Even after the work of Hennessy and Plotkin, powerdomains are not generally thought of as being *characterized* by their universal properties; but this is the view we will take here. After all, it is the universal property that matters most to the people who actually use (rather than study) powerdomains.

The convex or Plotkin powerdomain should be a kind of free algebra. Namely, let a **semilattice** be an object A equipped with a binary operation $*$: $A \times A \rightarrow A$ that is associative and commutative, and such that every element of A is idempotent. Let **Sl- $\mathcal{C}Sig$** be the category of semilattices whose underlying objects are complete Σ -spaces. The **convex powerdomain** functor $C: \mathcal{C}Sig \rightarrow \mathbf{Sl-}\mathcal{C}Sig$ should be the left adjoint to the forgetful functor $\mathbf{Sl-}\mathcal{C}Sig \rightarrow \mathcal{C}Sig$. The union map $\cup: CX \times CX \rightarrow CX$ would be the semilattice operation on the free algebra, and the singleton map $\{\}: X \rightarrow CX$ would be the unit of the adjunction.

Since $\mathcal{C}Sig$ is a small complete category, and **Sl- $\mathcal{C}Sig$** is monadic over $\mathcal{C}Sig$, this left adjoint exists for purely categorical reasons. And even this abstract description already has useful consequences; for example, powerdomains come equipped with associated modalities in a very natural way.

The point is that Σ is a semilattice under both $\wedge$ and $\vee$. So if $\phi: X \rightarrow \Sigma$ is a Σ -predicate of X , by the universal property of CX we get Σ -predicates $\Box\phi, \Diamond\phi: CX \rightarrow \Sigma$ of CX . For example, $\Box\phi$ satisfies

$$\begin{aligned}\Box\phi(\{x\}) &\longleftrightarrow \phi(x); \\ \Box\phi(t \cup t') &\longleftrightarrow \Box\phi(t) \wedge \Box\phi(t')\end{aligned}$$

and is characterized by those properties.

To prove finer results about CX we need a more explicit description of it, preferably a 'set-theoretic' one. But consider the diagram

$$\begin{array}{ccc} \mathbf{SI-CSig} & \longrightarrow & \mathbf{CSig} \\ \downarrow & & \downarrow \\ \mathbf{SI-Eff} & \longrightarrow & \mathbf{Eff} \end{array}$$

The vertical arrows are inclusions of full reflective subcategories and hence both monadic. Now the bottom arrow has a left adjoint that sends an object X to the object K^+X of finite nonempty subsets of X .

(We recall here that K^+X is the least subobject of Ω^X containing the singletons and closed under binary union. Another familiar description of K^+X would be as a set of equivalence classes of words. K^+X is not necessarily a Σ -space; in fact, it's almost never a Σ -space.)

This left adjoint lifts in a very straightforward way to a left adjoint C for the top arrow, by reflecting K^+X into $CSig$; this is our convex powerdomain functor. (There is no explicit description of CX , since there is at present no explicit description of the chain completion.)

[*Note.* We have to be careful here. What we really want—if we want to integrate powerdomains with dependent types—is for this left adjoint to exist in a fibred sense, over $\mathcal{E}ff$; but then we encounter the subtleties that were described in section 2.7.]

K^+X has another preorder on it in addition to its Σ -preorder $\leq$, namely the Egli-Milner preorder: if $s, t \in K^+X$ then $s \sqsubseteq_{EM} t$ iff

$$\forall x \in s \exists x' \in t. x \leq x' \quad \wedge \quad \forall x \in t \exists x' \in s. x' \leq x$$

where $\leq$ is the Σ -order on X .

The relationship between the Egli-Milner preorder and the Σ -preorder on K^+X has been studied in joint work with Paul Taylor; he first observed the following fact.

Proposition 9.1.1 (Taylor) $s \sqsubseteq_{EM} t$ implies $s \leq t$.

Note. In this section, only sketches of the proofs are given; a full account of the work, in a more general form due to Taylor, is in preparation: see [59].

Proof. (My version of a classical argument shown to me by Paul Taylor.) Firstly, s can be thought of as a word $x_1x_2\dots x_n (= \{x_1\} \cup \{x_2\} \cup \dots \cup \{x_n\})$ of elements of X . Similarly we can write $t = y_1y_2\dots y_m$ with $y_j \in X$.

Since $s \sqsubseteq_{EM} t$,

$$\forall x_i \in s. \exists x'_i \in t. x_i \leq x'_i$$

and

$$\forall y_j \in t. \exists y'_j \in s. y'_j \leq y_j.$$

But then

$$\begin{aligned} s &= x_1 \dots x_n y'_1 \dots y'_m \\ t &= x'_1 \dots x'_n y_1 \dots y_m \end{aligned}$$

by the semilattice laws. But since $\cup: K^+X \times K^+X \rightarrow K^+X$ is monotone in each argument (all maps are monotone with respect to the Σ -preorder), we must have $s \leq t$. ■

Corollary 9.1.2 *If X is a complete focal Σ -space, then so is CX .*

Proof. Let $\perp \in X$ be the bottom element of X ; then $\{\perp\}$ is a bottom element of K^+X under the Egli-Milner order and hence (by the proposition) also under the Σ -preorder. So by 7.1.3, its image under $K^+X \rightarrow CX$ is the bottom element of CX . ■

The following result says roughly that the Egli-Milner order and the Σ -order ‘look the same from a classical point of view’. It is due to Taylor and myself; there is a (harder) generalization due to Taylor.

Proposition 9.1.3 *In K^+X , $s \leq t$ implies $\neg\neg s \sqsubseteq_{EM} t$.*

Proof. We only give a sketch—namely, a ‘classical’ argument that $s \sqsubseteq_{EM} t$, which can be translated into an intuitionistically valid proof of $\neg\neg s \sqsubseteq_{EM} t$ in the internal logic. Note the use of choice over s and t , which is valid since countable choice holds in $\mathcal{E}ff$.

We’ll show that

$$[\neg\neg] \quad \forall x \in s. \exists x' \in t. x \leq x'.$$

(The proof of the other half is dual.)

Suppose $t = x'_1 \dots x'_n$, and fix $x \in s$. Suppose $\neg(x \leq x'_i)$ for all $i = 1, \dots, n$. Then

$$\forall i. \exists \phi_i \in \Sigma^X. \phi_i(x) \wedge \neg \phi_i(x'_i).$$

But if we let $\phi = \phi_1 \wedge \dots \wedge \phi_n$, then $\phi(x)$ but $\neg \phi(x'_i)$ for all i .

Now since Σ is a semilattice under $\wedge$, we get a Σ -predicate $\Diamond \phi \in \Sigma^{K^+X}$, namely

$$\Diamond \phi(x_1 \dots x_m) = \phi(x_1) \vee \dots \vee \phi(x_m).$$

And $\Diamond \phi(s)$ but $\neg \Diamond \phi(t)$, contradicting $s \leq t$. ■

[It is *not* trivial to turn this into a valid proof!] Taylor has announced a counterexample showing that the ' $\neg\neg$ ' cannot be removed.

Recall that equivalence relations in *Mod* are not effective, but that quotienting out by an equivalence relation is the same as quotienting out by its $\neg\neg$ -closure. The proposition tells us that the Σ -preorder and the Egli-Milner preorder induce the 'same' equivalence relation once we reflect into modest sets.

Incidentally the powerdomain functor, like the lift, is the object part of a monad—as we would expect from [47], where the classical powerdomain is cited as an example of a 'computational monad'.

Also note that if we require our semilattices to have identity elements—so that the free semilattice in *Eff* is the object KX of Kuratowski-finite subsets, rather than just K^+X , the inhabited ones—then we would get a powerdomain with the empty set as an isolated point, just as we do classically.

9.2 Modalities

This section sketches some logical properties of our convex powerdomain; it arises from work by Winskel, Robinson [64] and others on powerdomains and modalities (see also [78]).

One description of the classical Plotkin powerdomain CX is as follows: its frame of Scott-opens is generated by tokens $\Box \phi$, $\Diamond \phi$ (where ϕ runs through the Scott-opens of X) subject to the following relations:

1. $\Diamond \top = \Box \top = \top$.
2. If $\phi \leq \psi$ then $\Box \phi \leq \Box \psi$ and $\Diamond \phi \leq \Diamond \psi$.

3. If $\{\phi_i\}$ is directed then $\Box \bigvee \phi_i = \bigvee \Box \phi_i$.

4. $\Diamond \bigvee \phi_i = \bigvee \Diamond \phi_i$ for any family $\{\phi_i\}$.

5. $\Box(\phi \vee \psi) \leq \Box \phi \vee \Box \psi$.

6. $\Diamond(\phi \wedge \psi) \geq \Diamond \phi \wedge \Diamond \psi$.

I know of no such description of Σ^{CX} for our powerdomain; it seems that more powerful topological techniques must be developed before this would be possible. However, it is still of interest to know whether these logical laws hold.

Proposition 9.2.1 *1.–6. hold, provided that in 3. we restrict ourselves to chains $\{\phi_1 \leq \phi_2 \leq \dots\}$, and in 4. we restrict ourselves to countable families $\{\phi_1, \phi_2, \dots\}$.*

Proof. 2., 3. and 4. are general facts: 2. is true since

$$\Box, \Diamond: \Sigma^X \rightarrow \Sigma^{CX}$$

are monotone, 3. holds since $\Box$ preserves sups of chains, and 4. just says that binary and countable joins commute.

Since all the predicates involved are Σ -predicates, and $\Sigma^{CX} \cong \Sigma^{K^+X}$ (because Σ is a complete Σ -space and CX is the reflection of K^+X into $\mathcal{CSig}$), we may verify 1., 5. and 6. using the description of $\Box \phi, \Diamond \phi$ as maps $K^+X \rightarrow \Sigma$:

$$\begin{aligned} \Box \phi(x_1 \dots x_m) &= \phi(x_1) \wedge \dots \wedge \phi(x_m) \\ \Diamond \phi(x_1 \dots x_m) &= \phi(x_1) \vee \dots \vee \phi(x_m). \end{aligned}$$

Then 1., 5. and 6. are straightforward. ■

Paul Taylor has verified 1.–6. independently, in a general topos, for K^+X ; the proof is somewhat harder. Whether there are more general forms of 3. and 4. for CX is not clear to me.

9.3 Other constructions

The same idea leads us to more general algebraic domain constructions. For example, given X we might want to consider the nondeterministic multivalued computations of elements of X (cf. [79]); that is, the predomain $NM(X)$ with elements like

$$(x_1 \& x_2) \text{ or } (x_1 \& x_3 \& x_4).$$

This should be a semilattice under both $\&$ and or , and or should distribute over $\&$; so we can let $NM(X)$ be the free such thing. Again, it comes equipped with modality builders, and has an intuitive set-theoretic description.

Similarly, we could construct a very simple ‘process algebra on X ’; thinking of X as a set of actions or outputs, the algebra would have elements like

$$((x_1 \parallel x_2); x_3) \text{or} (x_3; (x_6 \text{or} x_8))$$

($\parallel$ is parallel output, $;$ is sequential output). But I’m not sure whether this ‘calculus of non-communicating systems’ will turn out to be very interesting.

If X is focal there is now no reason why (say) $NM(X)$ should also be focal. However, as long as we do not allow $NM(X)$ to contain the empty word, we can force $NM(X)$ to have a bottom element by identifying all words containing $\perp$. The operators $\&$, or etc. now all become *strict*.

One can make this rigorous as follows: we are given some equational theory Alg without constant symbols; let $\text{Alg-}CSig$ be the category of its models in $CSig$, $\text{Alg-}CFSig_{\perp}$ its models in $CFSig_{\perp}$.

It can be shown that the lift of an algebra in $CSig$ is an algebra in $CFSig_{\perp}$, that the lift operation is the functor part of a monad on $\text{Alg-}CSig$, and that the category of algebras for this monad is $\text{Alg-}CFSig_{\perp}$ —cf. 7.4.

Consider the diagram

$$\begin{array}{ccc} CFSig_{\perp} & \longleftarrow & \text{Alg-}CFSig_{\perp} \\ \downarrow & & \downarrow \\ CSig & \longleftarrow & \text{Alg-}CSig \end{array}$$

The vertical arrows are monadic. Also, $\text{Alg-}CFSig_{\perp}$ is small and complete and hence has coequalizers. So a left adjoint for the bottom arrow lifts to a left adjoint for the top arrow. For example, $NM: CSig \rightarrow CSig$ lifts to a new functor $NM_{\perp}: CFSig_{\perp} \rightarrow CFSig_{\perp}$.

If we unwind the proof of the relevant adjoint lifting theorem, we see that $NM_{\perp}(X)$ really is the quotient of $NM(X)$ in which all elements containing $\perp$ have been identified (it is obtained as a certain coequalizer).

Note that since $\wedge: \Sigma \times \Sigma \rightarrow \Sigma$ is strict but $\vee: \Sigma \times \Sigma \rightarrow \Sigma$ is not, we have a $\Box$ -type modality for $NM_{\perp}(X)$ but not a $\Diamond$ -type modality. In fact the logic of $NM_{\perp}(X)$, and of the other ‘strictified algebras’, seems quite weak.

9.4 Upper and lower powerdomains

The Plotkin powerdomain is only one powerdomain construction. It would also be useful to have a Smyth powerdomain functor $P_S: \mathcal{CSig} \rightarrow \mathcal{CSig}$ and a Hoare powerdomain functor $P_H: \mathcal{CSig} \rightarrow \mathcal{CSig}$. We sketch here the reason why it is just as trivial to construct these as it was to construct C .

There is an order-theoretic way of understanding the Smyth and Hoare powerdomains; that is, if one thinks of a domain X as being a poset, $P_S X$ should be the free meet-semilattice on X and $P_H X$ should be the free join-semilattice on X . But by 10.5.1, $\mathcal{CSig}$ is reflective in the category $\mathbf{Ord}$ of posets in $\mathcal{Eff}$; so to obtain $P_S X$ and $P_H X$, one simply constructs the free meet- or join-semilattice on X in $\mathbf{Ord}$ and reflects back into $\mathcal{CSig}$.

Note that the construction of free meet- and join-semilattices in $\mathbf{Ord}$ is quite concrete; we take KX (or K^+X , if we don't want a top/bottom) and impose the Smyth or Hoare preorder on it—one half or the other of the definition of the Egli-Milner order—then quotient out to obtain a partial order. It follows that finite subsets of X may be regarded as giving us (representatives of) certain elements of $P_S X$ or $P_H X$; and that the order on these particular elements of $P_S X$ or $P_H X$ is exactly the Smyth or Hoare preorder. This time there is no theorem to prove.

Since Σ is both a meet-semilattice (under the meet operation) and a join-semilattice (under the join operation), we have some corresponding modalities, and it is possible to prove modal laws for the Smyth and Hoare powerdomains in the same way as we did in 9.2; we don't bother to write them down here. But there are also other questions we might like to resolve; for example,

Conjecture 9.4.1 *The composite functors $P_H P_S$ and $P_S P_H$ from $\mathcal{CSig}$ to itself are naturally isomorphic.*

The intuition is that both of them give us the free distributive lattice functor (they do in $\mathbf{Ord}$). In [16], the corresponding theorem for the classical category of information systems is established; the proof there is quite involved. One would hope that in our setting one could find a more abstract proof. Incidentally, it seems that the best way to think of this result should be in terms of monads and distributive laws; but I haven't pursued this idea.

It seems as if the same approach should allow us to construct other kinds of powerdomains, so long as they have an algebraic characterization; for example,

there is a version of Gunter's *mixed powerdomain* construction: see [24]. Here the algebras are semilattices in the category of preorders, equipped with a unary operation that satisfies certain axioms. (But I'm not sure whether the resulting construction has the properties it's meant to have.)

This raises a number of interesting questions. For example, Gunter constructs the mixed powerdomain out of a binary relation between the upper and lower powerdomains (which I have not attempted to study in our context); and he also points out that, for example, no algebraic characterization has been found for the 'sandwiches' powerdomain used in the theory of databases—so here our simple-minded approach fails. I won't try to discuss these issues further.

Note. Some people have pointed out that there is a less 'synthetic' flavour about the constructions just described, as they are based on a partial order—and an imposed one, not an intrinsic one. This seems unavoidable if we insist on defining these powerdomains in an order-theoretic way. Another question is whether these constructions really give us the 'correct' and useful interpretations of nondeterminism which motivated the definitions of the classical powerdomains (such as the relationship between CX and bisimulation); to answer this, we ought to study them more closely and try to use them. I haven't begun to do this though.

Chapter 10

Topological questions

10.1 The Σ -doctrine

[*Note.* This chapter and the next consist mainly of observations, fragmentary (but suggestive) results, and conjectures. None of the topics discussed is properly understood at the moment, and a great deal of work remains to be done.]

In previous chapters—particularly in developing the basic theory of Σ -spaces in chapter 5—we have concentrated on the Σ -order on a Σ -space X . But (as Martin Hyland and Dana Scott have tried to point out to me many times) it is likely that the lattice Σ^X of Σ -subsets of X is fundamental—though (as we’ll see) we may have to change toposes for this idea to work out.

The Σ -subsets of X form a basis for a topology on X . And of course, any map $X \rightarrow Y$ is continuous with respect to the Σ -topologies on X and Y (see [68, proposition 3.4.1]).

As in classical domain theory (see for example [78]), there ought to be an intimate relationship between topology and logic: We regard a map $\phi \in \Sigma^X$ as being a computational property of elements of X . The fact that Σ is an ‘object of (semi)computable truth-values’ means that there is a natural ‘logic of Σ -spaces’ where the types are Σ -spaces, the terms are maps between Σ -spaces and the predicates are maps into Σ . In particular, the ‘effective’ or Σ_1^0 fragment of Floyd-Hoare logic has a natural interpretation inside *Sig*. Let’s recall some facts.

The logic of Σ includes quite a few logical operations:

$$\begin{aligned} \top, \perp &: 1 \rightarrow \Sigma \\ \wedge, \vee &: \Sigma \times \Sigma \rightarrow \Sigma \\ \exists a &: \Sigma^A \rightarrow \Sigma \end{aligned}$$

(where A is a closed quotient of $\mathbb{N}$). Furthermore, on certain ground types like $\mathbb{N}$ we have primitive Σ -predicates (like equality). And for most domain constructors, there are corresponding Σ -predicate builders.

For example, if $\phi \in \Sigma^X$ then since $X \subseteq_\Sigma X_\perp$, by 5.1.1 ϕ extends to a map $\phi_\perp \in \Sigma^{X_\perp}$:

$$\phi_\perp(t) \leftrightarrow (t \downarrow = x \wedge \phi(x)) \quad \text{'total correctness'}.$$

This lets us reason about possibly nonterminating programs.

However, note that if $\phi \in \Sigma^X$, the partial correctness predicate

$$t \downarrow = x \rightarrow \phi(x)$$

will not in general be a Σ -predicate; consider $\perp \in \Sigma^1$. I'm not sure how serious this problem is.

Also, recall that the powerdomains come equipped with modalities as a simple consequence of the way they are constructed. For instance, consider the convex powerdomain: if $\phi \in \Sigma^X$ then by the universal property of CX we get maps $\Box\phi, \Diamond\phi \in \Sigma^{CX}$; for example, $\Box\phi$ is characterized by

$$\begin{aligned} \Box\phi(\{x\}) &= \phi(x) & x &\in X \\ \Box\phi(s \cup s') &= \Box\phi(s) \wedge \Box\phi(s') & s, s' &\in PX. \end{aligned}$$

So we can reason about nondeterministic programs too.

In fact, Barney Hilken pointed out to me that any computational monad gives rise to associated modalities, so for purely formal reasons there are many predicate builders available to us.

Now if $\phi \in \Sigma^X$, $\psi \in \Sigma^Y$ and $f \in Y^X$ then the Hoare triple $\phi\{f\}\psi$ just says that $\phi \leq \psi f$ in Σ . And the usual logical rules are sound, e.g.

$$\begin{aligned} \text{if } \phi\{f\}\psi, \phi' \leq \phi, \psi \leq \psi' & \text{ then } \phi'\{f\}\psi'; \\ \text{if } \phi\{f\}\psi, \psi \leq \psi', \psi'\{f'\}\xi & \text{ then } \phi\{f'f\}\xi. \end{aligned}$$

I believe that this 'doctrine of Σ -predicates' is the natural setting in which to study the logic of effective domains and (higher-order, polymorphic) computable

functions. One should compare this idea with Abramsky's (classical) 'domain theory in logical form': see [2]. (Of course, Abramsky's programme has more mathematical substance and is quite fully developed.)

Unfortunately there is a snag; the 'topological' theory of Σ -spaces is poorly understood and seems very subtle compared to the 'order-theoretic' theory. In fact, as we will see at the end of this chapter, there is *no* full reflective subcategory of *Sig* which is 'topologically well-behaved'—at least not if we take a naive point of view.

That's why the next couple of chapters are so sketchy and inconclusive. Anyway, before we proceed, perhaps this is the place to make the obvious remark that full first-order logic, or even full propositional logic, is not available.

Proposition 10.1.1 $\Sigma^{\mathbb{N}}$ is not a Heyting algebra.

Proof. It suffices to show that the lattice of r.e. sets under inclusion is not a Heyting algebra. But if W is simple (i.e. if W is a coinfinite r.e. set meeting all infinite r.e. sets) then it's easy to see that W can have no recursively enumerable pseudocomplement. ■

10.2 Open spaces and maps

We have not mentioned finiteness conditions yet. It is somewhat against the spirit of the theory to talk about countable bases and so on; instead, we introduce an abstract finiteness condition, suggested by Martin Hyland, that is intimately related to the logic.

Call a map $f: X \rightarrow Y$ (logically) open if the 'image map'

$$\exists f: \Omega^X \rightarrow \Omega^Y$$

takes Σ -subsets to Σ -subsets; and call an object X open if the unique map $X \rightarrow 1$ is open. This means that if $\phi(x, \vec{y})$ is a computable (r.e.) predicate on X then so is $\exists x. \phi(x, \vec{y})$. (The term 'open' comes from the theory of locales.) For example, $\mathbb{N}$ is open, and any complete Σ -space with a countable dense subset—for instance, a countable basis—is open.

[Note that there are several other possible ways to define 'open'; I'm still not sure which is the right one. But certainly the results that follow should be valid for all of them, and are valid for the other ones that I can think of.]

The theory of open Σ -spaces is not well developed; only a few basic facts have been established.

Proposition 10.2.1 *If X is open, any quotient of X is open. Also, the chain completion X_G of X is open.*

Proof. Straightforward. ■

Proposition 10.2.2 (i) *If X and X' are open, then so is their product $X \times X'$.*

(ii) *If $\{X_n\}_{n \in \mathbb{N}}$ are open, then so is $\prod_n X_n$.*

(iii) *If X is open, so is the computable partial function space $[\mathbb{N} \rightarrow X]$.*

(iv) *If X is open and $X' \subset_\Sigma X$, then X' is open.*

(v) *If X is open then $X_\perp$ is open.*

Proof. (i) Suppose $\phi \in \Sigma^{X \times Y}$. For each $x \in X$ we have a Σ -predicate

$$\lambda y. \phi(x, y) \in \Sigma^Y.$$

So since Y is open, for each x we have

$$\exists y. \phi(x, y) \in \Sigma;$$

that is,

$$\lambda x. \exists y. \phi(x, y) \in \Sigma^X.$$

But X is open so

$$\exists x. \exists y. \phi(x, y) \in \Sigma.$$

(ii) First note that $\Sigma^X \cong \prod_n \Sigma^{X_n}$. For each $n \in \mathbb{N}$, let $\exists_n: \Sigma^{X_n} \rightarrow \Sigma$ be the left adjoint to $\Sigma \rightarrow \Sigma^{X_n}$; and let $\pi_n: \Sigma^X \rightarrow \Sigma^{X_n}$ be the projection. Then the left adjoint to $\Sigma \rightarrow \Sigma^X$ is

$$\sup_n \pi_n \exists_n.$$

(iii) Firstly, if $f \in [\mathbb{N} \rightarrow X]$ and $W = \text{dom } f$ then $W = \bigcup_n D_n$ where the D_n are a chain of finite, decidable subsets of $\mathbb{N}$; and

$$f = \sup_n f \upharpoonright D_n.$$

Hence if $\mathcal{U} \subset_\Sigma [\mathbb{N} \rightarrow X]$, then

$$f \in \mathcal{U} \longleftrightarrow \exists n. f \upharpoonright D_n \in \mathcal{U}.$$

So we may concentrate on the partial functions with finite (decidable) domains. Let $(D_i)_{i \in \mathbb{N}}$ be an enumeration of the finite subsets of $\mathbb{N}$ (given by canonical indices, say) and let d_i be the cardinality of D_i . We define

$$F_i \stackrel{\text{def}}{=} \{f \in [\mathbb{N} \rightarrow X] \mid \text{dom } f = D_i\} \cong X^{d_i}.$$

Then if $\mathcal{U} \subset_{\Sigma} [\mathbb{N} \rightarrow X]$,

$$\exists f. f \in \mathcal{U} \longleftrightarrow \exists f \in \bigcup_i F_i. f \in \mathcal{U}.$$

But $\bigcup_i F_i \cong \coprod_i X^{d_i}$ which is open by (i) and (ii).

(iv) By 5.1.1, any map $\phi: X' \rightarrow \Sigma$ extends to a map $\tilde{\phi}: X \rightarrow \Sigma$; and

$$\exists x' \in X'. \phi(x) \longleftrightarrow \exists x \in X. \tilde{\phi}(x) \wedge x \in X'.$$

(v) Suppose $U \subset_{\Sigma} X_{\perp}$ is classified by $\phi: X_{\perp} \rightarrow \Sigma$. ϕ is completely determined by its values on X and $\perp$, so

$$\exists t \in X_{\perp}. t \in U \longleftrightarrow (\emptyset \in U \vee \exists x \in X. x \in U)$$

which is an element of Σ since X is open. ■

For example, we can construct a solution X to the recursive domain equation

$$X \cong [\mathbb{N} \rightarrow (\mathbb{N} + X)]$$

that is an open complete Σ -space (form a certain countable coproduct of things of this form; take a quotient to get the colimit of a sequence of injections; reflect into $\mathcal{CSig}$, as noted in 6.2, what you get is isomorphic to the limit of a sequence of projections). That is, this domain of recursive records is an open Σ -space. Also,

Corollary 10.2.3 *If a complete Σ -space X is open, so is the convex powerdomain CX . If a complete focal Σ -space X is open, so is $NM_{\perp}(X)$.*

Proof. K^+X is a quotient of $\coprod_n X^n$ (think of it as consisting of equivalence classes of words) and hence open. But $\Sigma^{K^+X} \cong \Sigma^{CX}$ so CX must be open too.

$NM(X)$ is open by a similar argument. But $NM_{\perp}(X)$ is obtained from $NM(X)$ by taking a certain coequalizer in $\mathcal{CFSig}_{\perp}$; and by proposition 7.4.7, we may compute this coequalizer in $\mathcal{CSig}$. Now the coequalizer in $\mathcal{CSig}$ is the chain

completion of the coequalizer in Sig , which is a quotient of the coequalizer in Eff , which is a quotient of $NM(X)_\perp$; so, since the open objects are closed under lifting, quotients and chain completions, $NM_\perp(X)$ is open. ■

One might hope that if X and Y were open, then the partial function space $[X \multimap Y]$ would be open too. I have no proof of this; perhaps this is where algebraicity conditions have to enter, and I've so far avoided considering those. However, the following observations are trivial. See section 7.5 for an example of why they are useful.

Proposition 10.2.4 *Let $X, Y \in \text{CSig}$. If X is open, the complete focal Σ -space $[X \multimap Y]$ is a free algebra for the lift monad.*

Proof. By 7.4.6, it suffices to show that $D \subset_\Sigma [X \multimap Y]$, where

$$D = \{f: X \multimap Y \mid f \neq \perp\}.$$

But

$$D = \{f: X \multimap Y \mid \exists x. f(x) \downarrow\};$$

and since ' $f(x) \downarrow$ ' is a Σ -predicate on X and X is open, ' $\exists x. f(x) \downarrow$ ' is a truth-value in Σ for any f . ■

So, for example, $\Sigma^X \cong [X \multimap 1]$ will be a free algebra. The converse is also true.

Proposition 10.2.5 *If Σ^X is a free algebra for the lift monad, then X is open.*

Proof. Also easy, using 7.4.6. ■

As you can see, we really don't understand what's going on yet. A proper study of open objects (and, more generally, open maps) has not yet been carried out.

10.3 Topology, points and sobriety

Now we come to a problem which has tortured several people for a couple of years now. I wish I could offer a solution.

One could say that classical domain theory is really topological; or, if you don't like that, then at least that we spend a lot of time passing back and forth between the topological and order-theoretic points of view (*pace* Girard, who has his own ideas). And as usual, things work particularly well when we're dealing with *sober* spaces, which have a lot of nice properties. See for example [78].

So is there such a thing as a sober Σ -space? More precisely, is there a *useful* notion of sober Σ -space?

This problem was first tackled by Rosolini, in [68] and [69]; see section 14.2. Here I just want to put down a couple of definitions—based on a somewhat different philosophy—and make some observations.

The Σ -subsets on an object X generate a topology (in the usual sense) on X , which we'll call the Σ -**topology** and denote by $\mathcal{O}(X)$.

[There are indications that this is the wrong topology to consider; it is not 'algebraic' enough. Hyland has suggested that one should really only consider joins that exist 'uniformly', that is, the joins of I -indexed families of Σ -subsets where I is an object such that the map $\bigvee_i: \Sigma^I \rightarrow \Omega$ factors through Σ ; but that we should also consider more general infinitary algebraic operations. I don't know how to make use of these ideas. The definition above was chosen simply because it makes the theorems in the next chapter easy to state and prove; it should be regarded as a first approximation, especially since classical topological intuition is probably somewhat misleading.]

Note that if $\forall x. \{x\} \subset_{\Sigma} X$ then $\mathcal{O}(X) = \Omega^X$; for instance, this is what happens for $X = \mathbf{N}$. However, in general the Σ -topology is coarser than the discrete topology.

We won't need the following result in this chapter but we state it anyway.

Proposition 10.3.1 *The map $\mathcal{O}(X) \rightarrow \Omega^X$ is split mono.*

Proof. $\mathcal{O}(X) \cong (\Omega^X / \sim)$ where we define, for $X', X'' \in \Omega^X$,

$$X' \sim X'' \iff \forall U \subset_{\Sigma} X. (U \subseteq X' \leftrightarrow U \subseteq X'').$$

■

(For example, if $\{x\}$ is not a Σ -subset of X —e.g. $x = \perp \in \Sigma$ —then $\{x\} \sim \emptyset$.)

A **topological point** of X is a point of the frame $\mathcal{O}(X)$; this is the same thing as a map

$$\Sigma^X \rightarrow \Omega$$

that preserves binary meets and arbitrary unions (that is, any unions under which Σ^X is closed)—note that it must then preserve top and bottom. A **logical point** of X is a map

$$\Sigma^X \rightarrow \Sigma$$

that preserves binary meets and arbitrary unions. We let TX denote the object of topological points of X , and LX the object of logical points of X . Then obviously $X \mapsto LX \mapsto TX$.

Proposition 10.3.2 $T\Sigma \cong \Omega$ and $L\Sigma \cong \Sigma$.

Proof. As (both kinds of) points of Σ must be maps out of Σ^Σ preserving top and bottom, they are determined by what they do to $\text{id}_\Sigma \in \Sigma^\Sigma$. ■

Proposition 10.3.3 $T\mathbb{N} \cong L\mathbb{N} \cong \mathbb{N}$.

Proof. We prove this for logical points; the proof for topological points is the same. First, note that singletons are Σ -subsets of $\mathbb{N}$.

Let $p: \Sigma^\mathbb{N} \rightarrow \Sigma$ be a logical point of $\mathbb{N}$. It preserves the top element of $\Sigma^\mathbb{N}$, which is $\bigcup_n \{n\}$; so

$$\exists n. p(\{n\}) = \top.$$

What's more, if $p(\{n\}) = p(\{n'\}) = \top$ then

$$\top = p(\{n\}) \wedge p(\{n'\}) = p(\{n\} \cap \{n'\})$$

and hence $n = n'$. So

$$\forall p \in L\mathbb{N}. \exists! n \in \mathbb{N}. (\forall U \in \Sigma^\mathbb{N}. p(U) = |n \in U|);$$

that is, we have a map $L\mathbb{N} \rightarrow \mathbb{N}$; and this gives us an inverse for the natural mono $\mathbb{N} \rightarrow L\mathbb{N}$. ■

It's the logical points which look like being more interesting (unless we want to talk about toposes of sheaves; then, if we take the point of view of chapter 11, the topological points are the ones that crop up).

Clearly $X \mapsto LX \mapsto \Sigma^{\Sigma^X}$ in a natural way. Call X (**logically**) **sober** if the mono $X \mapsto LX$ is an isomorphism; for example, we've just seen that Σ and $\mathbb{N}$ are sober.

Proposition 10.3.4 *Every sober Σ -space is a complete Σ -space, but the converse does not necessarily hold.*

Proof. The first part is easy (cf. [69]); for the second part, we translate the example in [36] of a continuous poset whose Scott topology is not sober.

Fix an isomorphism $\mathbf{N} \cong \mathbf{N} \times (\mathbf{N} \cup \{\infty\})$ and consider the category whose objects are maps $f: \mathbf{N} \rightarrow X$ where X is a complete Σ -space and

$$\begin{aligned} f(m, n) \leq f(m', n') & \quad \text{if either } m = m' \text{ and } n \leq n' (\leq \infty) \\ & \quad \text{or } n' = \infty \text{ and } n \leq m' \\ f(m, \infty) &= \sup_n f(m, n) \end{aligned}$$

and whose morphisms are commutative triangles. Let $g: \mathbf{N} \rightarrow Z$ be the initial object of this category; it is the ‘universal image of Johnstone’s poset in a complete Σ -space’.

Z is not sober: Consider the subset $\mathcal{U} \subseteq_{\Sigma} \Sigma^Z$ defined by

$$\mathcal{U} = \{U \subseteq_{\Sigma} Z \mid \exists n \in \mathbf{N}. g(n) \in U\}.$$

An argument the same as the one in [36] shows that $\mathcal{U} \in LZ$ but that there is no $z \in Z$ such that $\mathcal{U} = \{U \subseteq_{\Sigma} Z \mid z \in U\}$. ■

[*Aside.* A similar construction shows that any nice countable chain-complete poset (in the category of sets) has a counterpart in $\mathcal{CSig}$; one should be able to say that the ‘computable’ continuous maps between two such posets correspond to the maps that exist between their counterparts in $\mathcal{Eff}$.]

We would like to prove that the full subcategory *Sob* of sober Σ -spaces is reflective. Since Σ is sober, we know what this reflection must be: if $S: \mathcal{Eff} \rightarrow \mathcal{Sob}$ is the reflection then

$$\Sigma^X \cong \Sigma^{SX}$$

and taking points gives us

$$LX \cong L(SX) \cong SX$$

since SX is sober.

Unfortunately there seems to be no simple proof that LX is sober. However, the following fact may be of interest.

Proposition 10.3.5 *Suppose X is a Σ -space and we have $X' \subset \Sigma^{\Sigma^X}$ such that $X \mapsto \Sigma^{\Sigma^X}$ factors through X' . If $\Sigma^X \cong \Sigma^{X'}$ (more precisely, if any map $X \rightarrow \Sigma$ extends uniquely to X') then $X' \subseteq LX$.*

(Cf. the discussion of sober spaces near [37, lemma 2.1].)

Proof. We will show that any $\mathcal{U} \in X'$, regarded as a map $\mathcal{U}: \Sigma^X \rightarrow \Sigma$, preserves binary meets. The other part is similar.

Let $U_1, U_2 \in \Sigma^X$. Define $F, G: \Sigma^{\Sigma^X} \rightarrow \Sigma$ by

$$\begin{aligned} F(\mathcal{U}) &= |U_1 \cap U_2 \in \mathcal{U}|; \\ G(\mathcal{U}) &= |U_1 \in \mathcal{U}| \wedge |U_2 \in \mathcal{U}|. \end{aligned}$$

Clearly F and G agree on the subobject $X \mapsto \Sigma^{\Sigma^X}$, since

$$(x \in U_1 \cap U_2) \longleftrightarrow (x \in U_1 \wedge x \in U_2).$$

Hence F and G agree on $X' \subset \Sigma^{\Sigma^X}$ as well. ■

10.4 Repletion

[The material described in this section is due to Martin Hyland; Paul Taylor has also done work along similar lines. It's included here partly because it's relevant to the present discussion, and partly because we'll need it in chapter 12. For full details see [30] and [77].]

At the moment, we don't know that *Sob* is reflective. But Hyland has shown that there is a smallest full reflective subcategory of *Eff* containing Σ ; it may turn out that this category is the same as *Sob*. In any case, it's clearly worth studying, though the definition is still rather abstract.

The idea is that to soberify a space, we 'add as many points to it as we can without changing the Σ -subsets'. The trick is to give some formal content to this intuition. The following definition makes sense for any dominance Σ .

Call a map $f: X \rightarrow Y \in \mathbf{Eff} \Sigma$ -anodyne if it induces an isomorphism $\Sigma^Y \rightarrow \Sigma^X$. If $f: X \rightarrow Y$ is Σ -anodyne, call f a Σ -anodyne extension of X . Call X replete if it is orthogonal to every Σ -anodyne extension. For example, by 10.3.5, any sober Σ -space is replete.

It is very easy to see that Σ is replete; this is almost immediate from the definition of Σ -anodyne extension. In fact, $\mathbf{N}$ is also replete; this follows from

the fact that it's sober, but also (as noted by Hyland) from the fact that the truth-values in Σ are closed under countable joins.

Proposition 10.4.1 (Hyland) *The category $\mathcal{R}ep$ of replete objects is a small complete full subcategory of $\mathcal{E}ff$; it is the smallest such subcategory that contains Σ (in fact, it's the reflective hull of Σ).*

The proof is omitted; it is abstract nonsense (but not trivial). There are also several other characterizations of $\mathcal{R}ep$ which are not given here.

Corollary 10.4.2 *Every replete object is a complete Σ -space.*

Proof. This follows from the fact that Σ is a complete Σ -space and that $\mathcal{C}Sig$ is a small complete full subcategory of $\mathcal{E}ff$. ■

Note the indirect nature of the proof. Because the definition of replete object is so abstract, this is usually the only way to prove results about replete objects.

To be able to do domain theory with $\mathcal{R}ep$, we also need the following result.

Proposition 10.4.3 (Hyland) *$\mathcal{R}ep$ is closed under the lift construction (that is, the Σ -partial map classifier defined in 7.2).*

The proof is easier than the proofs of the corresponding result for $\mathcal{C}Sig$; perhaps this is further evidence that $\mathcal{R}ep$ is the 'right' category. Anyway, the following result is evidence that replete objects really are well-behaved, and their elements and Σ -subsets are well related. The corresponding fact for complete Σ -spaces is far from obvious—in fact, it could well be false. (We won't be using it in the sequel, though.)

Proposition 10.4.4 (Hyland) *Let $X, Y \in \mathcal{R}ep$. Then a map $f: X \rightarrow Y$ is epi if and only if $\Sigma^f: \Sigma^Y \rightarrow \Sigma^X$ is mono.*

The point of view of Hyland—which is difficult to argue with—is that the replete objects are central, and that the only point of studying other, larger, categories is to prove things about $\mathcal{R}ep$ (as exemplified above).

Note that many (most?) of the techniques developed in the previous chapters can be used to study $\mathcal{R}ep$. Actually, I don't know of any other simple categorical properties that $\mathcal{R}ep$ enjoys which $\mathcal{C}Sig$ doesn't; though there are certainly complete

Σ -spaces which are not replete (see 14.3). It may turn out that for practical purposes, $\mathcal{CSig}$ is just as good as $\mathcal{Rep}$; certainly we don't yet understand the replete objects well enough to exploit their properties effectively.

By the way, it's not clear (to me) how to extend all of the results of chapter 8 to replete objects. Presumably this is possible.

10.5 An example in two models

There is a problem with the topological point of view that we've been trying to develop. Roughly speaking, it's that the categories of Σ -spaces that we've defined have well-behaved limits from an order-theoretic point of view, but not from a topological point of view. A pessimistic interpretation would be that the order-theoretic analysis works—but has its limits—whereas the topological analogy is more attractive, but flawed.

Let $\mathcal{Ord}$ be the category of partial orders (i.e. internal posets) and monotone maps in $\mathcal{Eff}$; and let $\omega\text{-}\mathcal{Ord}$ be the category of chain complete posets and monotone, chain-continuous maps in $\mathcal{Eff}$.

Both categories have small ($\mathcal{Eff}$ -indexed) limits—calculated by giving the set-theoretic limit the limit topology. But neither category is small (that is, neither category is the externalization of an internal category in $\mathcal{Eff}$). However,

Proposition 10.5.1 *$\mathcal{LSig}$ is reflective in $\mathcal{Ord}$.*

Proof. Since every map between linked Σ -spaces is monotone, there is a full embedding $I: \mathcal{LSig} \rightarrow \mathcal{Ord}$; and since limits of linked Σ -spaces have the pointwise order, I preserves and reflects all small limits. So, since $\mathcal{LSig}$ is essentially small, the result follows from the adjoint functor theorem. ■

And of course, the same holds for $\mathcal{CSig}$. In fact,

Proposition 10.5.2 *$\mathcal{CSig}$ is reflective in $\omega\text{-}\mathcal{Ord}$.*

Proof. The proof is similar; we get a full embedding because every map between complete Σ -spaces is monotone and preserves sups of chains, and from there the argument is the same. ■

(We are being imprecise again; really we're talking about the reflectivity of certain subfibrations of fibrations over Sep . And again, it's not too hard to make everything more rigorous.)

Now, let $\mathcal{S}p$ be the category of topological spaces and continuous maps in $\mathcal{E}ff$. As before, the forgetful functor $\mathcal{S}p \rightarrow \mathcal{E}ff$ reflects limits: if (X_i) is a diagram of spaces and $X = \varprojlim X_i$ is their set-theoretic limit, then we get the limit in $\mathcal{S}p$ by giving X the limit topology—that is, the topology generated by all sets of the form $\pi^{-1}U$, U open in X_i .

Let $\mathcal{C}$ be any full subcategory of $\mathcal{E}ff$ (for example, $\mathcal{C}Sig$ or $\mathcal{R}ep$). There is a functor $I: \mathcal{C} \rightarrow \mathcal{S}p$ that endows $X \in \mathcal{C}$ with the Σ -topology. (Recall that this is the topology on X generated by its Σ -subsets; it works on maps because the inverse image of a Σ -subset is a Σ -subset, and it's a full embedding.)

We'll show that if $\mathcal{C}$ contains $\mathbb{N}$ and is complete, then I can never have a left adjoint; we'll do this by showing that I doesn't reflect limits. More precisely: without loss of generality, $\mathcal{C}$ has countable products, and so contains $\prod_n \mathbb{N} = \mathbb{N}^{\mathbb{N}}$; but $\mathbb{N}^{\mathbb{N}}$ has the wrong Σ -topology.

Proposition 10.5.3 *The Σ -topology and the product topology on $\mathbb{N}^{\mathbb{N}}$ do not agree: the Σ -topology is strictly finer.*

The reason this is true is essentially because of [67, ex. 15–32(b), p370]. However, that result appears to be nonconstructive; so we have to be a little careful.

Lemma 10.5.4 *For any n , there exists a number k such that for no $e < n$ is it true that $\{e\}(0) = \dots = \{e\}(n-1) = 0$ and $\{e\}(n) = k$.*

Proof. This is simple; for example, we could take k to be greater than any value $\{e\}(i)$ for $0 \leq e < n$ and $0 \leq i \leq n$. However, note that this is not constructive, as there is no way of telling in advance whether a given computation $\{e\}(i)$ will converge. ■

A Σ -subset of $\mathbb{N}^{\mathbb{N}}$ may be identified with an r.e. set W such that if e and e' are indices for the same total recursive function f then $e \in W$ if and only if $e' \in W$; we call such an r.e. set **extensional**, and we can say that $f \in W$ without ambiguity. Also, we say that W has the **KLS property** if, whenever $f \in W$ then there exists a finite initial segment $\bar{f}$ of f such that if f' is any total recursive function

extending $\bar{f}$, then $f' \in W$. For example, the Kreisel-Lacombe-Shoenfield theorem says that every extensional recursive set has the KLS property. Also, the r.e. sets with the KLS property are clearly closed under unions.

Lemma 10.5.5 (Kreisel-Lacombe-Shoenfield) *Not every extensional r.e. set has the KLS property.*

Proof. Define an r.e. set W as follows—given a number e , run the following two tests on e in parallel:

- Test whether $\{e\}(0) = \dots = \{e\}(e) = 0$ (i.e. whether all of these computations converge and give the answer 0); and
- Compute $\{e\}(z)$ for $z = 1, 2, \dots$ one after the other; if you reach a z such that $\{e\}(z) \neq 0$ (so that $\{e\}(z') = 0$ for all $z' < z$) then test whether there exists an index $e' < z$ such that $\{e'\}(z') = \{e\}(z')$ for all $z' \leq z$.

If either succeeds, then $e \in W$.

It's easy to check that W is extensional; now suppose (for a contradiction) that W has the KLS property. Then since the everywhere zero function is in W , there is a number n such that if f is a total recursive function with $f(x) = 0$ for $x = 0, \dots, n-1$ then $f \in W$.

Let k be the number whose existence is assured by the previous lemma, and let g be the total recursive function defined by:

$$g(x) = \begin{cases} 0 & x \neq n; \\ k & x = n. \end{cases}$$

Then $g \in W$ by the definition of n ; but looking at the original definition of W , we see that $g \notin W$. ■

Proof. (Of proposition 10.5.3.) Let $\mathcal{O}$ be the Σ -topology on $\mathbb{N}^{\mathbb{N}}$, and let $\mathcal{O}'$ be the product topology; clearly $\mathcal{O}' \subseteq \mathcal{O}$. Let

$$\mathcal{U} = \{U \subset_{\Sigma} \mathbb{N}^{\mathbb{N}} \mid U \in \mathcal{O}'\};$$

it suffices to show that the inclusion $\mathcal{U} \subseteq \Sigma(\mathbb{N}^{\mathbb{N}})$ is strict.

Any r.e. subset of $\mathbb{N}$ is the union of decidable subsets; so any $U \in \mathcal{U}$ is the union of decidable subsets. That is, the global sections of $\mathcal{U}$ correspond to extensional

r.e. sets which are countable unions of extensional recursive sets; such r.e. sets must necessarily have the KLS property. So the extensional r.e. set W defined in the previous lemma corresponds to a global section of $\Sigma^{(\mathbb{N}^{\mathbb{N}})}$ which does not factor through $\mathcal{U}$. ■

There are at least two possible replies:

- a topological analysis might still somehow be possible—we just have to be more subtle; or
- things may be better in another model.

I have no ideas about the first point (though Hyland has proposed a new definition of ‘ Σ -locale’), and no evidence for the second. In fact the second suggestion seems unlikely to work; but the following result may be of interest (see chapter 12 for some definitions), though I’m not sure what its real significance could possibly be. It’s hard to see how to generalize it.

Proposition 10.5.6 *Suppose we’re working in the realizability topos over the r.e. graph model, and let Σ be the object defined in 12.3. Let $\pi_n: \mathbb{N}^{\mathbb{N}} \rightarrow \mathbb{N}$ be the product projections. Then given any $U \subset_{\Sigma} \mathbb{N}^{\mathbb{N}}$, there exist $U_{n1}, \dots, U_{nk_n} \subset_{\Sigma} \mathbb{N}$ such that*

$$U = \bigvee_n \bigwedge_{i=1, \dots, k_n} \pi_n^{-1} U_{ni}.$$

In fact, we can even take the U_{ni} to be singletons.

Proof. (Sketch.) Any $f \in \mathbb{N}^{\mathbb{N}}$ has a canonical representative that’s a set of elements of the form $\langle \{n\}, f(n) \rangle$.

Let $\phi: \mathbb{N}^{\mathbb{N}} \rightarrow \Sigma$ classify U . An element of the model representing ϕ must be an r.e. set of elements of the form

$$\langle \{ \langle \{ \dots \}, m_1 \rangle, \dots, \langle \{ \dots \}, m_k \rangle \}, 0 \rangle;$$

but as it acts on canonical representatives for the $f \in \mathbb{N}^{\mathbb{N}}$, it must have a representative containing countably many elements of the form

$$\langle \{ \langle \{n_1\}, m_1 \rangle, \dots, \langle \{n_k\}, m_k \rangle \}, 0 \rangle$$

(i.e. $f \in U$ iff $f(n_1) = m_1, \dots, f(n_k) = m_k$), for all such elements of the representative for ϕ . But then we may put $U_{ni} = \{m_i\}$. ■

It's a classical fact that combinatory algebras such as the graph models and function realizability models have better continuity properties than K_1 . Perhaps we'd have a better chance of finding a theory that's topologically well-behaved if we move out of $\mathcal{E}\mathcal{F}$ into one of these other realizability toposes.

Chapter 11

Sheaves

11.1 Definition and examples

In this chapter, Σ is any dominance which is a modest set in some realizability topos $\mathcal{E}$.

The idea of studying sheaves on objects of the form Σ^X is due to Fourman and Scott: see [74]. And it was Rosolini who suggested that I look at the *topos* of sheaves. Note that Scott's definition of 'sheaf' in [74] was different from the one given below: he only required the the glueing condition for countable covers. Cf. section 14.2.

Let X be any object of $\mathcal{E}$. Recall from 10.3 the definition of the Σ -topology $\mathcal{O}(X)$ on X . A *sheaf* on X is just a sheaf on this topological space.

We can give a more concrete description. Make Σ^X into a site by saying that a family of Σ -subsets (U_i) covers a Σ -subset U if

$$\bigcup_i U_i = U$$

(this might not be the canonical topology on Σ^X !); a sheaf on X is then a sheaf on this site.

In fact we can think about a sheaf (as Scott did) in a much more external way: as a family (F_{Σ^X}) together with an 'empty section' $* \in F$ over $\emptyset \subset_{\Sigma} X$ and a 'restriction map' $F \times \Sigma^X \rightarrow F$, satisfying certain conditions which we don't bother to spell out.

The category of sheaves on X is an $\mathcal{E}$ -topos, which we call $\text{Sh}(X)$. Let's look at a couple of examples of objects in $\text{Sh}(X)$.

The subobject classifier $\Omega_X \in \text{Sh}(X)$ is defined by

$$\Omega_X(U) = \{V \in \mathcal{O}(X) \mid V \subseteq U\}.$$

It has a subobject $\Sigma_X \in \text{Sh}(X)$ defined by

$$\Sigma_X(U) = \{V \subset_\Sigma X \mid V \subseteq U\}$$

(not to be confused with the constant sheaf $\Delta\Sigma$). It's easy to see that Σ_X is a dominance in $\text{Sh}(X)$.

More interesting examples are spaces of partial functions. Recall that there is a 'domain' map $[X \multimap Y] \rightarrow \Sigma^X$.

Proposition 11.1.1 *This gives us a sheaf on X , denoted (by slight abuse of notation) $[X \multimap Y] \in \text{Sh}(X)$, where*

$$[X \multimap Y](U) = \{f: X \multimap Y \mid \text{dom} f = U\}.$$

Proof. This is straightforward. To check the glueing condition, let $f_i: X \multimap Y$ be a collection of partial functions with $\text{dom} f_i = U_i \subset_\Sigma X$ and suppose that $U = \bigcup_i U_i \subset_\Sigma X$; suppose also that for all i, j ,

$$f_i|_{U_i \cap U_j} = f_j|_{U_i \cap U_j}.$$

Let

$$U' = \coprod_i \{\langle x, U_i \rangle \mid x \in U_i\};$$

then U is the image of the first projection $U' \rightarrow X$. Clearly we have a map

$$f' = \coprod_i f_i: U' \rightarrow Y.$$

We have to show that f' factors through $U' \twoheadrightarrow U$; that is, that it coequalizes the kernel pair of $U' \twoheadrightarrow U$. But if we calculate the kernel pair, we see that this is exactly the compatibility condition above. $\blacksquare$

In *Eff*, this has something to do with parallelism: If Σ is the r.e. subobject classifier, the union of two Σ -subsets is again a Σ -subset; so one consequence is that two partial functions which agree where they are both defined can be glued (interleaved). For example, it's possible to define a 'parallel-or' map $2_1 \times 2_1 \rightarrow 2_1$.

Scott was aware of some special cases of the following result (or rather, of its analogue for his definition of sheaf). In this setting it seems rather trivial; we won't be using it, and we mention it here mainly because it was discussed in [74].

Corollary 11.1.2 *If $X \in \mathcal{E}$, the convergence predicate $X_1 \rightarrow \Sigma$ displays X_1 as a sheaf over 1, whose object of global sections is X .*

Proof. Obvious since $X_1 = [1 \rightarrow X]$. ■

One more example might be of interest. For any objects X, Y define $D \in \text{Sh}([X \rightarrow Y])$ by

$$D(U) = \{x \in X \mid \forall f \in U. x \in \text{dom } f\}.$$

We'll say more about this sheaf in the next section.

11.2 Sections and stalks

We want to construct a geometric morphism $s: \mathcal{E}/X \rightarrow \text{Sh}(X)$ with the interpretation

$$\begin{aligned} s_* &= \text{sheaf of sections,} \\ s^* &= \text{stalk space.} \end{aligned}$$

We can do this as follows.

Sheaf of sections: If $(\frac{T}{X}) \in \mathcal{E}/X$ define $s_*T \in \text{Sh}(X)$ by

$$s_*T(U) = \{f: X \rightarrow T \mid \text{dom } f \subseteq_\Sigma X, pf = 1 \text{ on } \text{dom } f\}.$$

Stalk space: If $F \in \text{Sh}(X)$ define $s^*F \in \mathcal{E}/X$ to be the X -indexed family of objects $(s^*F)_x$ where

$$(s^*F)_x = \lim_{\rightarrow U \ni x} F(U).$$

Proposition 11.2.1 *This defines a surjection of toposes.*

Proof. The two constructions evidently describe functors—they are defined on maps in the obvious way. The proof that these functors are adjoint is the same as the proof (for ordinary topological spaces) that the ‘stalk space’ functor and the ‘sheaf of sections’ functor are adjoint. Similarly, s^* is left exact because finite limits commute with filtered colimits, and so with $\lim_{\rightarrow U \ni x}$. Finally, the fact that s is a surjection is a special case of proposition 11.5.1. ■

Some notation: If F is a sheaf on X we will denote s^*F by $(F_x)_{x \in X}$; informally, we call F_x the **stalk** of F at x ; and if $U \ni x$ and $f \in F(U)$, we'll denote the corresponding element of F_x by $[f]$.

For example, the stalks of the sheaf D defined at the end of the last section are

$$D_f = \text{dom } f.$$

A degenerate example of the operation of s_* on a badly behaved ('discontinuous') X -indexed family: Consider the case $X = \bar{\omega}$. The mono $\omega \mapsto \bar{\omega}$ gives rise to an object $(\frac{\omega}{\bar{\omega}}) \in \mathcal{E}/X$, and

$$s_*(\frac{\omega}{\bar{\omega}}) = 0 \in \text{Sh}(X).$$

There is no reason to expect s to be an *open* surjection. And also, s is not usually essential; in fact,

Proposition 11.2.2 *s^* need not preserve countable products.*

Proof. Let $(F_n)_{n \in \mathbb{N}}$ be a countable family of sheaves in $\text{Sh}(X)$.

First consider $(\frac{T}{X}) = \Pi_n s^* F_n$, the product of their stalk spaces; the stalks are

$$T_x = \text{sequences } ([f_n]) \text{ with } [f_n] \in (F_n)_x$$

Note that we may have representatives f_n with different domains (shrinking as n increases, say).

Now consider $(\frac{S}{X}) = s^* \Pi_n F_n$, the stalk space of their product; the stalks are

$$S_x = \text{equivalence classes of sequences } (f_n), f_n \in F_n(U), U \ni x$$

where all the f_n are understood to have the same domain.

In general, $S_x \mapsto T_x$ is strict (though if we were talking about a *finite* product, and Σ^X were closed under meets, then they would be isomorphic).

For example, let $\Sigma =$ (the r.e. subobject classifier in $\mathcal{E}\text{ff}$) and let $F_n = [[\mathbb{N} \multimap \mathbb{N}] \multimap \mathbb{N}]$ for all n ; let $X = [\mathbb{N} \multimap \mathbb{N}]$, let $x = 1_{\mathbb{N}} \in X$ and let $f_n: [\mathbb{N} \multimap \mathbb{N}] \multimap \mathbb{N}$ be defined by

$$\begin{aligned} \text{dom } f_n &= \{g: \mathbb{N} \multimap \mathbb{N} \mid \{1, \dots, n\} \in \text{dom } g\} \\ f_n(g) &= g(1) + \dots + g(n) \end{aligned}$$

Then $([f_n]) \in T_x$ but there does not exist $U \subset_{\Sigma} [\mathbb{N} \multimap \mathbb{N}]$ with $x \in U \subseteq \text{dom } f_n$ for all n . (This is because

$$\bigcap_{n=1}^{\infty} \text{dom } f_n = \mathbb{N}^{\mathbb{N}}$$

but by the Myhill-Shepherdson theorem, there are no nonempty Σ -subsets of $[\mathbb{N} \multimap \mathbb{N}]$ which contain only total functions.) ■

11.2.1 A more concrete construction of stalks

The definition of the stalk space relied heavily on the internal logic; in fact, there is a more external version which is not too complicated. Here it is. It's still not completely external, but the remaining traces of the internal language are just cosmetic, and it's not hard to get rid of them—just messy.

If $F \in \text{Sh}(X)$ let $(\frac{F}{\Sigma^X})$ be the corresponding object of $\mathcal{E}/\Sigma^X$; taking the exponential transpose gives us a map

$$\begin{aligned} X &\rightarrow \Sigma^F \\ x &\mapsto \{f \mid x \in \text{dom} f\}. \end{aligned}$$

Define $(\frac{E}{\Sigma^F})$ as the composite

$$E = \in \mapsto F \times \Sigma^F \rightarrow \Sigma^F$$

where $\in$ is the membership relation and the second arrow is the product projection (E is the generic Σ -subset of F : if $V \in \Sigma^F$,

$$E_V = \{\langle f, V \rangle \mid f \in V\}.$$

Also, note that the restriction operation for sections gives us an action $\Sigma^X \times F \rightarrow F$.)

Form the pullback

$$\begin{array}{ccc} D & \longrightarrow & E \\ \downarrow & & \downarrow \\ X & \longrightarrow & \Sigma^F \end{array}$$

so that

$$D_x = \{\langle x, f \rangle \mid x \in \text{dom} f\}.$$

Now in the fibre over $x \in X$ define

$$Q_x = \{U \subset_\Sigma X \mid x \in U\};$$

then we have an action $Q_x \times D_x \rightarrow D_x$ over X , sending

$$\langle U, \langle x, f \rangle \rangle \mapsto \langle x, f \upharpoonright U \cap \text{dom} f \rangle.$$

Form the pullback

$$\begin{array}{ccc} C_x & \longrightarrow & D_x \\ \downarrow & & \downarrow \\ Q_x \times D_x \times D_x & \longrightarrow & D_x \times D_x \end{array}$$

(where the right-hand vertical arrow is the diagonal map) so that

$$C_x = \{\langle U, \langle x, f \rangle, \langle x', f' \rangle \rangle \mid f \upharpoonright U = f' \upharpoonright U\};$$

and let $(s^*F)_x$ be the coequalizer of the pair of maps obtained by composing

$$C_x \mapsto Q_x \times D_x \times D_x \rightarrow D_x \times D_x$$

(the second map is a product projection) with the two product projections $D_x \times D_x \rightarrow D_x$. That is, we're identifying $\langle x, f \rangle$ and $\langle x, f' \rangle$ if f and f' agree on some $U \ni x$.

Incidentally, if the $F(U)$ are nice (e.g. $\neg\neg$ -separated, or Σ -spaces, or something) then there's no reason to expect the stalks $(s^*F)_x$ to be as nice; forming the coequalizer will probably take us outside any good category we happened to be in.

11.2.2 Local homeomorphisms?

Since s is a surjection of toposes, $\text{Sh}(X)$ is the topos of coalgebras for a certain left exact comonad on $\mathcal{E}/X$. This is really a special case of a classical fact: given a topological space X , there is a left exact comonad on Sets/X whose topos of coalgebras is LH/X , the category of local homeomorphisms into X (which is equivalent to $\text{Sh}(X)$, of course).

This suggests that there might be a notion of 'local homeomorphism' appropriate to Σ -spaces. But I haven't been able to find one. Well, one could just define it to be a coalgebra for this comonad, but that doesn't seem to be a very useful thing to do. So I don't know.

11.3 Continuously dependent types

Let X be a data type; we think of a dependent type over X as being an X -indexed collection of types

$$T(x) \quad [x \in X].$$

If $\mathcal{C}$ is our category of types, then a dependent type is traditionally modelled by an object $\begin{pmatrix} T \\ X \end{pmatrix} \in \mathcal{C}/X$.

But what if our objects are domains? Then we might want the semantics of dependent type to take into account the notions of information ordering and

approximation. That is, if $x \leq x'$ we'd like to be able to compare $T(x)$ and $T(x')$; and if $x = \sup_n x_n$, we'd like to be able to say that $T(x) = \varinjlim T(x_n)$. But if we're just modelling dependent types by objects of slice categories, it's not clear how to make sense out of these things.

They do make sense if we model dependent types as sheaves—that is, as *continuously* varying sets. That is, a continuously dependent type over X should be a sheaf $T \in \text{Sh}(X)$. The type $T(x)$ for a particular value of x is just the stalk T_x of T at x . (Or, to put it more parametrically, one can obtain from T an ordinarily dependent type over X —that is, an object of $\mathcal{E}/X$ —by applying the stalk space functor $s^*: \text{Sh}(X) \rightarrow \mathcal{E}/X$.)

Suppose $x \leq x'$. Now x and x' both give rise to points $\mathcal{E} \rightarrow \text{Sh}(X)$; and the stalks T_x and $T_{x'}$ are just the inverse images of T . But since $x \leq x'$, we get a natural transformation between the points $\mathcal{E} \rightrightarrows \text{Sh}(X)$, and hence a comparison map

$$T_x \rightarrow T_{x'}.$$

Proposition 11.3.1 *If $x = \sup x_n$, then $T_x = \varinjlim T_{x_n}$.*

Proof. Let $Y \in \mathcal{E}$, and suppose we have a family of maps $T_{x_n} \rightarrow Y$ commuting with the maps $T_{x_n} \rightarrow T_{x_{n+1}}$.

Recall that by definition,

$$T_{x_n} = \varinjlim_{U \ni x_n} T(U)$$

(where $T(U)$ is the object of sections of T over $U \subseteq_\Sigma X$). So for all $U \subseteq_\Sigma X$ containing x_n we have a map

$$f_{n,U}: T(U) \rightarrow Y;$$

and if $m > n$, $f_{m,U} = f_{n,U}$.

Now suppose we have $U \subseteq_\Sigma X$ with $x \in U$. Then by 5.3.3,

$$\exists n. x_n \in U$$

and so we have a map $T(U) \rightarrow Y$; this map is independent of the choice of n such that $x_n \in U$. But since

$$T_x = \varinjlim_{U \ni x} T(U),$$

we get a map $T_x \rightarrow Y$. And uniqueness follows similarly. ■

What's going on in this proof? Suppose we have a continuously dependent type $T(x)$ over X (that is, a sheaf $T \in \text{Sh}(X)$) and an element $x \in X$. If we only have a finite amount of information about x —if we know that $x \in U$ for some $U \subset_\Sigma X$ —then we still know something about $T(x) = T_x$, namely that it's approximated by $T(U)$. (Though in T_x some of the elements of $T(U)$ may become identified, and new elements may appear.)

Note. Taking the 'propositions-as-types' point of view, and thinking of a dependent type over X as a predicate on X , the proposition says roughly that a sheaf over X can be regarded as an inclusive predicate. I haven't followed up this idea.

11.3.1 A sheaf model for continuously dependent types

So what we'd like is to use sheaf toposes to give a semantics for theories of dependent types. Here's an indication of how things might go.

[*Warning.* The 'results' below are all conjectural; they depend on an assumption, spelled out later, for which I have no evidence. So I don't claim that there is anything substantial here, even by the standards of the rest of the chapter; all I have is a collection of suggestive ideas which, I think, deserve to be pursued further.]

First a bit of suggestive notation. Instead of writing $\text{Sh}(X)$ for the topos of sheaves on X , we'll write $\mathcal{E}[x \in X]$; the intuition is that $\mathcal{E}[x \in X]$ is obtained from $\mathcal{E}$ by adjoining a generic, continuously varying element of type X . And we'll now denote a typical object of $\mathcal{E}[x \in X]$ by $T(x)$. (The clash of notation is deliberate, I'm afraid.)

The basic idea: If X is a type—which we model as an object of the topos $\mathcal{E}$ —then we model a dependent type

$$T(x) \quad [x \in X]$$

(a syntactic object) by some sheaf

$$T(x) \in \mathcal{E}[x \in X].$$

Iterated contexts: $\Sigma \in \mathcal{E}$ gives rise to a dominance $\Delta\Sigma \in \mathcal{E}[x \in X]$; so if $Y(x) \in \mathcal{E}[x]$, it's possible to define the $\mathcal{E}[x \in X]$ -topos

$$\mathcal{E}[x \in X, y \in Y(x)]$$

of sheaves on $Y(x)$. Then a ‘doubly’ dependent type

$$T(x, y) \quad [x \in X, y \in Y(x)]$$

will be modelled by a sheaf

$$T(x, y) \in \mathcal{E}[x \in X, y \in Y(x)].$$

(Incidentally, all the definitions below which are formulated over $\mathcal{E}$ can be similarly relativized; they still make sense in nontrivial contexts. We only leave out the more general formulations for simplicity.)

Reindexing: Let X and Y be types, let

$$T(y) \quad [y \in Y]$$

be a dependent type over Y , and suppose

$$f(x) \in Y \quad [x \in X]$$

is a term of type Y with a free variable in X . The interpretation of f will be an arrow $X \rightarrow Y \in \mathcal{E}$; this gives rise to a geometric morphism $f: \mathcal{E}[x \in X] \rightarrow \mathcal{E}[y \in Y]$. Then if

$$T(y) \in \mathcal{E}[y \in Y]$$

is the interpretation of the dependent type $T(y) [y \in Y]$, the interpretation of the reindexed dependent type

$$T(f(x)) \quad [x \in X]$$

will be the inverse image

$$f^*T(y) \in \mathcal{E}[x \in X]$$

—which we may as well denote $T(f(x))$.

So far, it all works well. But to model product and sum types, it seems as if we need an extra assumption: namely, that **everything is locally connected**. More precisely, we’ll have to assume that for any object X under consideration, the geometric morphism

$$\mathcal{E}[x \in X] \rightarrow \mathcal{E}$$

is locally connected. This means that the inverse image functor has an $\mathcal{E}$ -indexed left adjoint Λ ; that is, we have

$$\Lambda \dashv \Delta \dashv \Gamma$$

where Δ is the constant sheaf functor and Γ is the global sections functor. That's a pretty strong assumption and might not be consistent; but this is probably only a first approximation to a semantics anyway.

The first attractive consequence is that pullbacks are what we'd expect them to be. Namely, let $X, Y \in \mathcal{E}$; then

$$\begin{array}{ccc} \mathcal{E}[x \in X, y \in \Delta Y] & \longrightarrow & \mathcal{E}[x \in X] \\ \downarrow & & \downarrow \\ \mathcal{E}[y \in Y] & \longrightarrow & \mathcal{E} \end{array}$$

is a pullback of toposes. This follows from the fact that one can form the pullback of a bounded geometric morphism by pulling back the site, and the fact that Δ preserves exponentials: $\Delta(\Sigma^Y) \cong \Delta\Sigma^{\Delta Y}$ (and a calculation shows that the site structure on the two is the same as well).

Product and sum types: Since we used inverse image functors to model reindexing, we are more or less forced to model product and sum types by direct image functors and by the left adjoints to inverse image functors respectively. For example, if the interpretation of a dependent type $T(x) [x \in X]$ is a sheaf $T(x) \in \mathcal{E}[x \in X]$, then the interpretation of

$$\Pi x \in X. T(x) \quad []$$

(empty context) is just

$$\Gamma T(x) \in \mathcal{E};$$

and similarly for sum types. The required Beck condition for Π 's works since pullbacks along locally connected geometric morphisms satisfy the Beck condition; and the Beck condition for Σ 's follows in the usual way by taking adjoints.

So everything ought to drop out neatly, and it should turn out that all the hard work's already been done by the topos theorists. What a pity it's not that easy.

Closing remarks. As for the local connectedness assumption: There ought to be conditions on the dominance Σ ensuring that each Σ^X gives rise to a locally connected site (which is certainly the case if $\Sigma = \Omega$, rather trivially). But I haven't a clue what form such conditions would take.

There is another problem anyway. Consider the case $\mathcal{E} = \mathcal{E}ff$ for the moment; then $\Delta\Sigma \in \text{Sh}(X)$ is actually the 'r.e. subobject classifier' (the closure of $2 = \Delta 2$ under countable joins) in $\text{Sh}(X)$; so the notions discussed in the past few chapters

also make sense in $\text{Sh}(X)$. We want to be able to talk about continuously varying families of Σ -spaces, complete Σ -spaces and so on; that is, Σ -spaces etc. in the toposes $\text{Sh}(X)$. In fact, ideally we'd like to know that

Conjecture 11.3.2 *$S \in \text{Sh}(X)$ is a Σ -space (or whatnot) in $\text{Sh}(X)$ if and only if each $S(U)$ is a Σ -space in $\mathcal{E}$.*

The relativized versions should hold too, of course. Furthermore, one would like various constructions to be stable under reindexing. Some of these things should not be too hard to check; but clearly there is some left to do here.

11.4 Specification spaces

Let's start this section by recalling the theory of constructions as described, for example, in [32] (see also the discussion of a weaker system in chapter 4); in particular, I want to discuss the intuitive distinction between *kinds* and *types*—or at least, one view of kinds vs. types inspired by the models that arise in realizability toposes such as $\mathcal{E}ff$. The following remarks are extremely informal; and perhaps they should be taken with a grain of salt.

The kinds we are interested in are typically built up from the kind of all types; this is because we're thinking of type polymorphism as quantification over such kinds. In $\mathcal{E}ff$, the interpretation of the kind of all types is the object-of-objects of some internal category, and we're not interested in its computational structure; indeed, it's often a $\neg\neg$ -sheaf and thus has no 'computational structure' (e.g. Mod_0).

On the other hand, we are interested in the internal structure of types—for instance, a type such as $[\mathbb{N} \rightarrow \mathbb{N}]$ is not just a discrete collection of partial functions; it has a Σ -order, a Σ -topology, and we may want to take these things into account. $[\mathbb{N} \rightarrow \mathbb{N}]$ is something we compute with, whereas (for example) CSig_0 isn't.

Let me illustrate this idea by talking about packages. Roughly speaking, a package ought to be something that

- Accepts some parameters which could be

types $T_1, \dots, T_n$

objects/terms $x_1 : X_1, \dots, x_k : X_k$, each with a given type (and presumably the types X_i of the object parameters depend on the type parameters T_i)

- Spits out a packet of **components**, namely some

types $S_1, \dots, S_m$

objects/values $y_1 : Y_1, \dots, y_l : Y_l$, each with a given type (that depends on the S_i , which in turn depend on the T_i)

- Satisfies certain **requirements**, embodied in predicates

$$\phi_1(y_1), \dots, \phi_l(y_l)$$

where each ϕ_i may have $x_1, \dots, x_k$ as free variables (parameters).

(But note that this is not quite the same picture as EML has, for example.)

What would be a semantics for such a specification? The S_i are presumably uniquely determined by the T_i (whose domain of variation is a kind) in a purely syntactic way; they should be objects in

$$\mathcal{E}/(T_1 \times \dots \times T_n).$$

But there might be many possibilities for each y_i ; so we want to consider the collection of *all* suitable ‘solutions’ $\langle y_1, \dots, y_l \rangle$ (that is, all the ones for which the ϕ_i hold). And we think of the y_i as being computed from the x_i (whose domain of variation is a type). Taking this view, it’s reasonable to consider solutions which are only partly defined, and to expect to be able to glue different, but compatible, partially defined solutions together. So the family of all the $\langle y_1, \dots, y_l \rangle$ should be an object of

$$\text{Sh}(X_1 \times \dots \times X_k)$$

(which is an $\mathcal{E}/(T_1 \times \dots \times T_n)$ -topos).

For example, the sheaf $[X \rightarrow Y] \in \text{Sh}(X)$ can be thought of as the space of (partial) solutions to the problem

‘(given $x \in X$) construct an element $y \in Y$ ’.

Or if $p: T \rightarrow X$ is some map, the sheaf $s_*\left(\frac{T}{X}\right) \in \text{Sh}(X)$ can be thought of as the space of partial solutions to the problem

‘(given $x \in X$) construct an element $t \in T$ such that $p(t) = x$ ’.

It might seem that we're only talking about partial correctness, as we allow the solutions to be only partially defined on the X_i ; however, we could obtain the totally defined solutions by taking global sections.

11.4.1 Defining predicates

If we're going to use sheaves to give a semantics for specifications, we'd better know something about the internal logic of these sheaf toposes. Let's just see how to define some basic predicates on sheaves. Recall from 11.1 the definition of the subobject classifier $\Omega_X \in \text{Sh}(X)$, as well as $\Sigma_X \mapsto \Omega_X$, the object of truth-values with extent a Σ -subset of X .

Given $V \subseteq_\Sigma X$ we obtain a global section $\ulcorner V \urcorner$ of Σ_X corresponding to the elements

$$\ulcorner V \urcorner_U = V \cap U \in \Sigma_X(U);$$

so if $S \in \text{Sh}(X)$, we have a constant predicate $S \rightarrow \Sigma_X$ sending $f \mapsto \ulcorner V \urcorner_U$; it classifies $S|_V \mapsto S$ where

$$S|_V(U) = S(V \cap U).$$

We could call it 'restriction to V '.

In fact, any predicate $\psi: X \rightarrow \Omega$ in $\mathcal{E}$ gives rise to a global section $\ulcorner \psi \urcorner \in \Omega_X$, namely the one corresponding to the elements

$$\ulcorner \psi \urcorner_U = \{V \subseteq U \mid V \subseteq X'\} \in \Omega_X(U)$$

(where $X' \mapsto X$ is the subobject classified by ψ).

Also, if $S = s_*(\frac{T}{X})$ for some object $(\frac{T}{X}) \in \mathcal{E}/X$, then predicates on $(\frac{T}{X})$ give rise to predicates on S just by applying s_* to their extensions (remember that s_* preserves monos). That is, if we have an X -indexed collection of subobjects

$$T'_x \mapsto T_x,$$

the corresponding predicate on F turns out to classify the largest subsheaf $F' \mapsto F$ for which

$$F'_x \mapsto T'_x$$

for all x ; it is the 'best continuous counterpart' of a condition which has been defined pointwise (i.e. for each x individually).

11.5 Different dominances

All the results so far hold for an arbitrary dominance. If we have two different dominances, what's the relation between the toposes of sheaves that we get?

Let $\Sigma_1 \subset \Sigma_2$ be two dominances. If $X \in \mathcal{E}$, let $\mathcal{O}_1(X)$, $\mathcal{O}_2(X)$ be the topologies on X generated by its Σ_1 -subsets and its Σ_2 -subsets, and let $\text{Sh}_1(X)$, $\text{Sh}_2(X)$ be the corresponding toposes of sheaves.

Proposition 11.5.1 *There is a surjection of toposes $\text{Sh}_2(X) \rightarrow \text{Sh}_1(X)$.*

Proof. It's induced by the continuous map of internal topological spaces

$$\langle X, \mathcal{O}_2(X) \rangle \rightarrow \langle X, \mathcal{O}_1(X) \rangle,$$

which is surjective (indeed, the identity) on underlying 'sets' (objects of $\mathcal{E}$): cf. [35, 4.12(iv)]. ■

And everything is natural. So to sum up, a dominance gives rise to an $\mathcal{E}$ -indexed localic $\mathcal{E}$ -topos (that is, a family of $\mathcal{E}$ -toposes and geometric morphisms between them indexed by the objects and arrows of $\mathcal{E}$); and inclusions of dominances give rise to $\mathcal{E}$ -indexed surjections. The picture is something like

$$\begin{array}{ccccccc}
 & & \mathcal{E}/Y & \longrightarrow & \mathcal{E}/X & & \\
 & \nearrow & \downarrow & & \downarrow & \searrow & \\
 1 & \longrightarrow & \text{Sh}_2(Y) & \longrightarrow & \text{Sh}_2(X) & \longrightarrow & \mathcal{E} \\
 & \searrow & \downarrow & & \downarrow & \nearrow & \\
 & & \text{Sh}_1(Y) & \longrightarrow & \text{Sh}_1(X) & & \\
 \\
 0 & \longrightarrow & Y & \longrightarrow & X & \longrightarrow & 1
 \end{array}$$

where the vertical arrows are surjections.

11.6 Gros toposes

An ordinary sheaf topos arises from a space. A *gros topos* arises from a category of spaces, and an object of a gros topos can be regarded as a kind of 'polymorphic sheaf'. In this section we briefly look at how this works in the context of Σ -spaces.

There is nothing new here. Most of what follows is just a paraphrase of the account in [6]—the infamous *SGA4*—and it's only reproduced here for convenience.

(But note that since we are dealing with small categories, the foundational difficulties that the authors of [6] had to deal with don't arise here.)

Let $\mathbf{C}$ be one of the small complete categories we've been considering—**Mod**, **Sig**, **CSig**, **Rep**—for the moment it doesn't matter. We can regard $\mathbf{C}$ as a site by letting the covering families be jointly epi families of inclusions of Σ -subsets. (The paper [42] explains what 'site' means in an arbitrary topos, for readers unwilling to take this for granted.) Let $\text{Sh}(\mathbf{C})$ be the topos of sheaves on this site.

More generally, if $X \in \mathbf{C}$ then the forgetful functor $\mathbf{C}/X \rightarrow \mathbf{C}$ induces a topology on $\mathbf{C}/X$; the corresponding topos of sheaves is called the *gros topos* of X , denoted $\text{SH}_{\mathbf{C}}(X)$ or just $\text{SH}(X)$. Clearly

$$\mathbf{C}/X \rightarrow \text{SH}(X)$$

is fully faithful. (In fact,

$$\text{SH}(X) \simeq \text{Sh}(\mathbf{C})/yX$$

where $y: \mathbf{C} \rightarrow \text{Sh}(\mathbf{C})$ is the Yoneda embedding.)

The inclusion $\Sigma^X \rightarrow \mathbf{C}/X$ induces a local geometric morphism

$$\text{SH}(X) \rightarrow \text{Sh}(X)$$

(see [38]). In particular, we have geometric morphisms

$$\begin{aligned} f: \text{Sh}(X) &\rightarrow \text{SH}(X) \\ g: \text{SH}(X) &\rightarrow \text{Sh}(X) \end{aligned}$$

such that $gf \cong id_{\text{Sh}(X)}$.

The functor $g_* = f^*: \text{SH}(X) \rightarrow \text{Sh}(X)$ is just the obvious restriction functor (given a presheaf $\mathbf{C}/X \rightarrow \mathcal{E}^{op}$, compose with $\Sigma^X \rightarrow \mathbf{C}/X$; this clearly takes sheaves to sheaves).

We'll now give a more concrete description of $\text{SH}(X)$ and use it to define g^* . Let $F \in \text{SH}(X)$ and let $(\frac{X'}{X}) \in \mathbf{C}/X$ (so that $\text{Sh}(X')$ is a $\text{Sh}(X)$ -topos). Then F restricts to an ordinary sheaf $F_{X'} \in \text{Sh}(X')$. If

$$u: (\frac{X''}{X}) \rightarrow (\frac{X'}{X})$$

is a map in $\mathbf{C}/X$ (inducing a geometric morphism $\text{Sh}(X'') \rightarrow \text{Sh}(X')$ over $\text{Sh}(X)$); then we get a transition map

$$\phi_u: u^* F_{X'} \rightarrow F_{X''}.$$

And given another map

$$v: \left(\frac{X'''_1}{X}\right) \rightarrow \left(\frac{X''_1}{X}\right),$$

it's easy to see that $\phi_{uv} = (v^* \phi_u) \phi_v$.

Also observe that if u is an open inclusion $X'' \hookrightarrow X'$ —i.e. an inclusion of some member of $\mathcal{O}(X')$ —then ϕ_u is an isomorphism. Conversely, given a system

$$\begin{aligned} F_{X'} &\in \text{Sh}(X') && \text{for each } \left(\frac{X'_1}{X}\right) \in \mathbb{C}/X \\ \phi_u: u^* F_{X'} &\rightarrow F_{X''} && \text{for each } u: \left(\frac{X''_1}{X}\right) \rightarrow \left(\frac{X'_1}{X}\right) \in \mathbb{C}/X \end{aligned}$$

satisfying these conditions, we can reconstruct a sheaf $F \in \text{SH}(X)$.

Now we can describe $g^*: \text{Sh}(X) \rightarrow \text{SH}(X)$. Given $S \in \text{Sh}(X)$ it associates to each $\left(\frac{X'_1}{X}\right) \in \mathbb{C}/X$ the sheaf

$$p^* S \in \text{Sh}(X')$$

and to each map $u: \left(\frac{X''_1}{X}\right) \rightarrow \left(\frac{X'_1}{X}\right)$, the isomorphism

$$\phi_u: u^* p^* S \rightarrow q^* S$$

(remember that $q = pu$). The essential image of g^* consists of those $F \in \text{SH}(X)$ for which all the transition maps ϕ_u are isomorphisms. We think of such a sheaf F as begin ‘constant as X' varies’.

For example, if $S \in \text{Sh}(X)$ is the sheaf of partial functions $[X \rightarrow Y]$ for some fixed object Y and $F = g^* S$ then

$$F_{X'} = \{h: X' \rightarrow Y \mid h \text{ factors through } p: X' \rightarrow X\}.$$

On the other hand, the sheaf $F \in \text{SH}(\Sigma)$ defined by

$$F_{X'} = \{h: X' \rightarrow \mathbb{N} \mid h \uparrow \text{ outside } p^{-1}(\top)\}$$

(for $\left(\frac{X'_1}{\Sigma}\right) \in \mathbb{C}/\Sigma$) is not in the essential image of g^* .

11.6.1 So what?

The reason that gros toposes might be interesting is that they could provide a model for polymorphic specifications or polymorphic dependent types—that is, specifications or dependent types with unspecified domain of variation.

Consider $\text{SH}(1) = \text{Sh}(\mathbb{C})$. To give an object F of this topos, we have to give a compatible family of sheaves $F_X \in \text{Sh}(X)$ for each $X \in \mathbb{C}$ (where ‘compatible’

means that we have appropriate transition maps); that is, a ‘polymorphic sheaf’. And the inverse image of the composite geometric morphism

$$\mathrm{Sh}(X) \rightarrow \mathrm{SH}(X) \rightarrow \mathrm{SH}(1)$$

corresponds to instantiation at $X \in \mathcal{C}$.

For example, we could let

$$F_X = [X \rightarrow \mathbb{N}]$$

with the obvious transition maps; this is the ‘polymorphic sheaf of $\mathbb{N}$ -valued partial maps’. Instantiation at any particular object X yields the ordinary sheaf $[X \rightarrow \mathbb{N}]$.

Perhaps it’s even possible to use this idea to extend our (conjectural) model of continuously dependent types into a model for the theory of constructions, with a gros topos modelling the kind of types. And what about polymorphic inclusive predicates, and polymorphic invariance?

11.6.2 A problem with the site sig

Recall the full embedding $\mathcal{C}/X \rightarrow \mathrm{SH}(X)$. It tells us that certain dependent types over X can be thought of as sheaves in the corresponding gros topos.

The nice thing about the modest sets is that by proposition 8.1.1, they are closed under coproducts indexed over themselves. What this means is that if X is modest then any X -indexed family of modest sets gives us an object of $\mathbf{Mod}/X$, and hence, a sheaf in the gros topos for which $\mathbf{Mod}/X$ is the site.

The same remark applies to the category $\mathcal{L}$ defined at the end of section 5.2. But by the result of Streicher noted at the beginning of section 8.2, it does not apply to Sig . That is, not all families of Σ -spaces indexed by a given Σ -space X are represented by objects of $\mathrm{SH}(X)$.

11.6.3 Open surjections?

In algebraic geometry we consider several kinds of topologies on a given category: not just the Zariski topology, but (for example) those arising from étale morphisms, or fppf morphisms. Is it possible to do something like that here?

A positive answer to this question could have some practical applications. For example, suppose there were a good notion of ‘open surjection’. Then an open

surjection $Y \rightarrow X$ would induce an open surjection of toposes $\text{Sh}(Y) \rightarrow \text{Sh}(X)$; that is, the logic of specifications over Y would be ‘conservative over’ the logic of specifications over X . So we could think of the data type Y as being a faithful representation of the data type X . (All this was discussed at more length, and somewhat prematurely, in [52].)

Unfortunately, there is a snag. The problem is that I don’t know of a definition of open map that must be stable under pullback; so the jointly epi families of ‘open’ maps, as we can presently define them, won’t necessarily form a Grothendieck topology on our categories. Yet again, our poor understanding of the Σ -topology is an obstacle to progress.

(Interestingly enough, the same problem arises in algebraic geometry: there is no guarantee that arbitrary open maps are stable under pullback. And the lazy solution—to look at universally open maps, i.e. maps that remain open after arbitrary base extension—is no good either, because there’s no concrete characterization of the universally open maps. Flatness was the solution there; is this evidence that a slick new idea might be able to solve *our* problem too?)

Chapter 12

The graph model

12.1 Calculating indexed products

The material in this and the next chapter is not yet in a satisfactory state. One problem is that the definitions involve explicit references to PERs: not a very abstract approach. (Some of the categories we define may not even be closed under isomorphism!) Then to work with these definitions, we must resort to calculating with PERs. Even when this is not too difficult, there's still the feeling that we could gain more insight from an abstract treatment.

The following result lets us avoid a few calculations when we're trying to show that some class of modest sets is closed under indexed products.

Proposition 12.1.1 *Let C_0 be a $\neg\neg$ -closed subobject of $\mathbf{Mod}_0$; think of it as defining a full subfibration C of Mod (over Sep). Then to show that C is closed under indexed products along all arrows in Sep , it suffices to show that*

- C is an exponential ideal; and
- if $(R_i)_{i \in I}$ is a family of PERs in C_0 indexed by a $\neg\neg$ -sheaf I then their intersection $\bigcap_{i \in I} R_i \in C_0$.

Note. The proof is basically the same as the argument appearing in [28]. The result implies closure under indexed products (in a weak sense, cf. 2.5) along arbitrary maps of the topos.

Proof. (Sketch; section numbers refer to [28].) We can conclude from the second condition that C is closed under indexed products along all maps of $\neg\neg$ -sheaves (that is, thought of as fibred over $Sets$, it's closed under indexed products).

Since C_0 is a $\neg\neg$ -closed subobject of a $\neg\neg$ -sheaf, and hence itself a $\neg\neg$ -sheaf, any map $X \rightarrow C_0$ determines (as in 2.5) a diagram

$$\begin{array}{ccccc} R & \longrightarrow & R' & \longrightarrow & R'' \\ \downarrow & & \downarrow & & \downarrow \\ X & \longrightarrow & sX & \longrightarrow & aX \end{array}$$

(where sX is the reflection of X into Sep , and aX is its $\neg\neg$ -sheafification), in which both squares are pullbacks.

Now apply the argument of 2.7. ■

Of course, checking the hypotheses still involves *some* stuffing around with PERs.

12.2 The r.e. graph model

We'll quickly review the Plotkin-Scott construction of graph models for the λ -calculus; see [73] or [9] for more details.

Let X be a set equipped with a coding of pairs $\langle \cdot, \cdot \rangle: X \times X \rightarrow X$ and a coding of finite subsets $\text{Fin}X \rightarrow X$. Let PX be the powerset of X . We can define a binary application operation $PX \times PX \rightarrow PX$ as follows: if $X_1, X_2 \subseteq X$, put

$$X_1 X_2 = \{x \in X \mid \exists y \in X. (y \subset X_2) \wedge (\langle y, x \rangle \in X_1)\}$$

where ' $y \subset X_2$ ' means that $y \in X$ codes a finite subset of X_2 .

This—together with a few other definitions which are omitted here—turns PX into a model for the untyped λ -calculus. We call it a *graph model*; submodels of such models are also called graph models.

From now on, if $y \in X$ codes a finite subset $\{x_1, \dots, x_n\}$, we will abuse notation and simply write $\{x_1, \dots, x_n\} \in X$ in place of y . (But note that different codings of pairs or finite subsets will, in general, give rise to different models; so this is actually a more serious abuse than it looks.)

The natural numbers come equipped with a standard (bijective) pairing $\mathbb{N} \times \mathbb{N} \rightarrow \mathbb{N}$ as well as a coding of finite subsets $\text{Fin}\mathbb{N} \rightarrow \mathbb{N}$:

$$\{n_1, \dots, n_k\} \mapsto \sum_{i=1}^k 2^{n_i}.$$

So $P\mathbb{N}$ becomes a model for the untyped λ -calculus.

It has a submodel consisting of just the r.e. subsets; we will call this the **r.e. graph model** (or, in this chapter, just ‘the graph model’) and denote it by R . This is the model that we’ll be looking at.

12.2.1 Why look at the graph model?

After all, I said we were interested in term models.

The first reason is that the graph model is a lot easier to work with. Term models are not very tractable—and there’s always the risk that one’s going to get caught up in the theory of reduction, which would be a disaster—whereas working with R feels much more like ordinary mathematics.

But secondly, the r.e. graph model is a sort of term model, or at least a hard model: not for the pure λ -calculus, but for Scott’s language LAMBDA. (This is, roughly speaking, the λ -calculus plus strong definition by cases—see [73]. If you prefer combinators, Beeson’s language PCA is essentially the same thing: [9].)

And LAMBDA is a very natural language to consider. It’s very closely related to Plotkin’s PCF; it contains some arithmetic and choice, and then the full machinery of the λ -calculus; thought of as an abstract machine language, it is almost practical.

But, one technical point which is certainly important. What equality on terms is forced by R ?

We know that two λ -expressions denote the same element of a graph model precisely when they have the same Böhm tree (see [8, chapter 19] for an exposition). But, as Beeson points out, the corresponding problem for LAMBDA doesn’t seem to have been studied. One would expect it to be similarly related to some kind of deep operational equivalence. Recent unpublished work of Kennaway, Klop, Sleep and de Vries deals with a notion of Böhm tree for general infinitary term rewriting systems; this may suggest a way to start tackling the problem.

12.2.2 Relative dominances

We will need a slightly more general notion of dominance and lifting. Let $\mathcal{C}$ be a subcategory of $\mathcal{E}$, such as *Sep* or *Mod*; then it is clear what it means for an object $\Sigma \rightarrow \Omega$ to be a **dominance relative to $\mathcal{C}$** :

- for any X , $X \subset_{\Sigma} X$; and

- if $X \subset_{\Sigma} Y$ and $Y \subset_{\Sigma} Z$ then $X \subset_{\Sigma} Z$.

where X, Y, Z range over the objects of $\mathcal{E}$.

If $\mathcal{C}$ comes from an internal category $\mathbf{C}$ in $\mathcal{E}$, these conditions can be expressed in the internal language of $\mathcal{E}$. If not, we ask that they hold in an appropriately fibred sense.

12.3 Partial maps and lifting

As in $\mathcal{E}ff$, we begin by describing a suitable notion of computable partial map; equivalently, by defining a dominance. The idea will be that the element $\emptyset \in \mathbf{R}$ represents ‘divergence’ or ‘no result’.

Let Σ be the $\mathbf{R}$ -valued assembly with underlying set $\{\top, \perp\}$ and

$$|\perp| = \emptyset, \quad |\top| = \{0\}.$$

This gives us an object Σ in the realizability topos $\mathcal{E}$; in fact, a modest set. There are several other ways to define Σ ; note that it’s the quotient of the following (internally defined) $\neg\neg$ -closed PER R on $\mathbf{R}$, the object of realizers in $\mathcal{E}$ that comes from $\mathbf{R}$:

$$pRq \iff \neg(p = q = \emptyset) \rightarrow (p = q = \{0\}).$$

In fact, $\Sigma \mapsto \nabla 2$; as a consequence, if $X' \subset_{\Sigma} X$ and $x \in X'$ then ‘codes for x in X' ’ are the same thing as codes for x in X ’; this will simplify our arguments.

Proposition 12.3.1 *Σ is a dominance relative to $\mathbf{Mod}$.*

Note. The following proof, like most of the proofs in this chapter, is going to look rather external. The rigorous internal versions would be basically the same, but with lots of extra λ -abstractions (since we would be establishing the validity of universally quantified formulas instead of particular instances). That is, in the external version we typically start with some r.e. sets (elements of $\mathbf{R}$) and construct other r.e. sets from them; in the internal version (which talks about elements of the corresponding object $\mathbf{R} \in \mathcal{E}$ and internally defined operations on them), we’d have to explicitly effectivize the construction by exhibiting an appropriate enumeration operator. Well, this would be tedious, so we won’t do it.

Proof. All we really have to show is that if $X \subseteq_{\Sigma} Y$ and $Y \subseteq_{\Sigma} Z$ then $X \subseteq_{\Sigma} Z$. Suppose $p \in \mathbf{R}$ tracks the classifier $Y \rightarrow \Sigma$ of X , and $q \in \mathbf{R}$ tracks the classifier $Z \rightarrow \Sigma$ of Y ; then there is a map $Z \rightarrow \Sigma$ tracked by

$$\{ \langle \{m_1, \dots, m_k\} \cup \{n_1, \dots, n_l\}, 0 \rangle \mid \langle \{m_1, \dots, m_k\}, 0 \rangle \in q, \langle \{n_1, \dots, n_l\}, 0 \rangle \in p \}$$

that classifies $X \mapsto Z$. ■

Now we'd like an explicit description of the lift of a modest set. First note that there's a predicate $\mathbf{R} \rightarrow \Sigma$ tracked by

$$\{ \langle \{0\}, 0 \rangle \}$$

that classifies the $\neg\neg$ -closed subobject of $\mathbf{R}$ corresponding to

$$\{p \in \mathbf{R} \mid 0 \in p\} \subset \mathbf{R}.$$

Also, we'll need the fact that the operation

$$p \mapsto p - 1 = \{n - 1 \mid n \in p\}$$

is tracked by an element of $\mathbf{R}$, namely

$$\{ \langle \{n + 1\}, n \rangle \mid n \in \mathbf{N} \}.$$

Finally, recall that the operation of binary union is definable by an element of $\mathbf{R}$.

Let R be a PER and define a new PER $R_{\perp}$ by

$$pR_{\perp}q \iff \neg(p = q = \emptyset) \rightarrow ((p - 1)R(q - 1) \wedge 0 \in p \wedge 0 \in q).$$

(Observe that this is indeed a $\neg\neg$ -closed partial equivalence relation.) Then

Proposition 12.3.2 *If $Y \cong \mathbf{R}/R$, then $Y_{\perp} \cong \mathbf{R}/R_{\perp}$.*

Proof. First note that $Y \subseteq_{\Sigma} \mathbf{R}/R_{\perp}$; its classifying map is tracked by $\{ \langle \{0\}, 0 \rangle \}$.

Let $X' \subseteq_{\Sigma} X$ and suppose p tracks a map $X' \rightarrow Y$ and q tracks the classifier $X \rightarrow \Sigma$ of X' . Then the corresponding map $X \rightarrow \mathbf{R}/R_{\perp}$ is tracked by

$$q \cup \{ \langle \{m_1, \dots, m_k\} \cup \{n_1, \dots, n_l\}, r + 1 \rangle \mid \langle \{m_1, \dots, m_k\}, r \rangle \in p \wedge \langle \{n_1, \dots, n_l\}, 0 \rangle \in q \}.$$

Conversely, given a map $X \rightarrow \mathbf{R}/R_\perp$, we can obtain a partial map $X \multimap Y$ with domain a Σ -subset by pulling back along $Y \subset_\Sigma \mathbf{R}/R_\perp$. ■

Observe that $R_\perp$ has the property that $\emptyset R_\perp \emptyset$. We will call such PERs **canonically focal**. (This notation will be explained in the next section.) If a modest set is the quotient of such a PER, we will call it a **canonically focal** modest set. Note that the canonically focal modest sets need not be closed under isomorphism.

Proposition 12.3.3 *The fibration of canonically focal sets is closed under indexed products.*

Proof. Immediate from the fact that canonically focal sets form an exponential ideal and that canonically focal PERs are closed under arbitrary intersection: cf. [28, 2.7]. (To see that the canonically focal sets form an exponential ideal, observe that if X is canonically focal then for any object Y , $\emptyset$ tracks a map $Y \rightarrow X$.) ■

12.4 A partial order

It seems difficult to do anything with the Σ -order here; at least, it's more convenient to use the link relation $\triangleleft$, defined in sub-section 5.4.1.

The theme of this chapter and the next is that properties of the underlying combinatory algebra are reflected in suitable PERs over it. Now the inclusion relation on $\mathbf{R}$ gives us a $\neg\neg$ -closed relation

$$\triangleleft \mapsto \mathbf{R} \times \mathbf{R}.$$

Lemma 12.4.1 *This relation coincides with the link relation on $\mathbf{R}$.*

Proof. First suppose that we have $p, q \in \mathbf{R}$ with $p \triangleleft q$; that is, there exists $f: \Sigma \rightarrow \mathbf{R}$ with $f(\perp) = p$, $f(\top) = q$. Now f is tracked by some $a \in \mathbf{R}$; so it suffices to show that

$$\forall a \in \mathbf{R}. ((a\emptyset = p \wedge a\{0\} = q) \rightarrow p \triangleleft q).$$

Note that this formula is $\neg\neg$ -stable, so 'it has the same meaning as it does in *Sets*' (see [27]). And it's certainly valid there.

Conversely, we must show that if $p \triangleleft q$ then $p \triangleleft q$; that is, we must construct a suitable map

$$\triangleleft \rightarrow \mathbf{R}^\Sigma,$$

or, which comes to the same thing, a map

$$\triangleleft \times \Sigma \rightarrow \mathbf{R}.$$

Now in $\mathbf{R} \times \mathbf{R}$ we have

$$|\langle p, q \rangle| = \{2n | n \in p\} \cup \{2n+1 | n \in q\};$$

so the codes for elements of $\triangleleft \rightarrow \mathbf{R} \times \mathbf{R}$ are r.e. sets r such that

$$2n+1 \in r \longrightarrow 2n \in r.$$

So codes for elements of $\triangleleft \times \Sigma$ are r.e. sets of the form

$$\{2m | m \in r\}$$

or

$$\{2m | m \in r\} \cup \{1\};$$

(where r is an r.e. set satisfying the above condition) and the required map $\triangleleft \times \Sigma \rightarrow \mathbf{R}$ is tracked by

$$\{\{\{4n\}, n\} | n \in \mathbf{N}\} \cup \{\{\{4n+2, 1\}, n\} | n \in \mathbf{N}\}.$$

■

If $X \cong \mathbf{R}/R$ is a modest set, consider the relation $\triangleleft$ on X defined by

$$x \triangleleft x' \longleftrightarrow \exists p, p' \in \mathbf{R}. ([p]_R = x \wedge [p']_R = x' \wedge p \triangleleft p').$$

(Note that this definition only makes sense once we have fixed a particular R such that $X \cong \mathbf{R}/R$.) By the lemma, this coincides with the link relation. So the link relation on a modest set has a very natural interpretation.

Observe that not all modest sets are link-ordered. For example, consider the modest set with underlying set $\{a, b, c\}$ (three arbitrary symbols) and

$$|a| = \{\{0\}\}, \quad |b| = \{\{0, 1\}, \{2\}\}, \quad |c| = \{\{2, 3\}\};$$

then $a \triangleleft b$ and $b \triangleleft c$ but not $a \triangleleft c$.

We showed in sub-section 5.4.1 that the category of objects X for which the link relation was a partial order was a full reflective subcategory of $\mathcal{M}od$. This will be our substitute for the category $\mathcal{LSig}$ in $\mathcal{E}ff$.

Call a link-ordered object **focal** if it has a bottom element. Again using the lemma, it's easy to see that any canonically focal link-ordered object is focal. But note that, while canonically focal objects are injective with respect to Σ -subsets, there's no obvious reason why focal objects should be.

Proposition 12.4.2 *For any modest set X , $X_\perp$ is focal.*

Proof. Immediate from the explicit description of $X_\perp$ given above. ■

However, if X is link-ordered, there is no reason to suppose that $X_\perp$ should be: as an external fact about PERs on $\mathbf{R}$, this seems obvious, but the constructive proof breaks down (try it!).

12.5 Chain completeness and fixed points

What will replace $\mathcal{CSig}$? Consider the category of link-ordered objects for which the link order has sups of chains (in an internal sense—to labour the point again). Since the link order on a limit of such objects is pointwise, it's easy to see that this category is again small and complete. But this is not enough; we want to show that if X and Y are objects of this category then any map $f: X \rightarrow Y$ preserves sups of chains. And there seems to be no reason why this should be true.

To prove such a continuity principle, we need to move to a smaller category. First, let's take a closer look at what chains should be. Let ω be the $\mathbf{R}$ -valued assembly with underlying set $\{0, 1, 2, 3, \dots\}$ and

$$|0| = \emptyset, |1| = \{1\}, |2| = \{1, 2\}, |3| = \{1, 2, 3\}, \dots$$

This gives rise to an object ω of $\mathcal{E}$. We would like to think of this object as the 'generic chain'.

Indeed, if X is a link-ordered object, then maps $\omega \rightarrow X$ give rise to chains in X . But the converse is not true; there are 'bad' objects X in which we can find chains which do not come from maps $\omega \rightarrow X$.

For example, let X have underlying set $\{0, 1, 2, 3, \dots\}$ and put

$$\begin{aligned} |0| &= \{\{0\}, \quad \{1\}, \quad \{2\}, \quad \{3\}, \dots\} \\ |1| &= \{\{0, 1\}, \quad \{1, 2\}, \quad \{2, 3\}, \quad \{3, 4\}, \dots\} \\ |2| &= \{ \quad \quad \{1, 2, 3\}, \quad \{2, 3, 4\}, \quad \{3, 4, 5\}, \dots\} \\ |3| &= \{ \quad \quad \quad \{2, 3, 4, 5\}, \quad \{3, 4, 5, 6\}, \dots\} \\ |4| &= \{ \quad \quad \quad \quad \{3, 4, 5, 6, 7\}, \dots\} \end{aligned}$$

The elements of X clearly form an infinitely ascending chain; but the chains of their *codes* all terminate after a finite number, so there can be no map $\omega \rightarrow X$ corresponding to this chain.

To see how to exclude this and other pathologies, we need a few more definitions. Let $\bar{\omega}$ be the assembly with underlying set $\{0, 1, 2, 3, \dots, \infty\}$ with $|n| = \{1, 2, \dots, n\}$ as above and

$$|\infty| = \{1, 2, 3, \dots\}.$$

It will be the ‘generic chain with a supremum’. Also, let $\bar{\omega}^+$ be the assembly with underlying set $\{0, 1, 2, 3, \dots, \infty, \top\}$ with $|n| = \{1, 2, \dots, n\}$ and $|\infty| = \{1, 2, \dots\}$ as before, and

$$|\top| = \{0, 1, 2, 3, \dots\}.$$

It will be the ‘generic chain with (a supremum and) an upper bound’.

Lemma 12.5.1 *Any map out of $\bar{\omega}^+$ is uniquely determined by its action on $1, 2, 3, \dots$ and on $\top$.*

Proof. By $\neg\neg$ -density, a map out of $\bar{\omega}^+$ is uniquely determined by its action on $1, 2, 3, \dots, \infty$ and $\top$; but by continuity of application in the graph model, its action on $1, 2, 3, \dots$ determines its action on ∞ . ■

Let X be link-ordered. Call X a **uniformly complete modest set** if for every chain (x_n) in X

- there exists a map $f \in X^{\bar{\omega}^+}$ such that $f(n) = x_n$ and $f(\infty) = f(\top)$; and
- for every upper bound $x \in X$ for (x_n) , there exists a map $f \in X^{\bar{\omega}^+}$ such that $f(n) = x_n$ and $f(\top) = x$.

By the lemma, these maps must necessarily be unique.

Proposition 12.5.2 *In the situation just described, $f(\infty) = \sup_n x_n$. Furthermore, if $g: X \rightarrow Y$ is any map from X to another uniformly complete modest set, then*

$$g(\sup_n x_n) = \sup_n g(x_n).$$

Proof. The first statement follows easily from the characterization of the link relation in terms of the inclusion relation, and the monotonicity of application in the graph model.

As for the second statement: Let the chain (x_n) give rise to a map $f: \bar{\omega}^+ \rightarrow X$ with $f(\infty) = f(\top)$; then

$$g(x_n) = g(f(n)) \triangleleft g(f(\infty)) = g(\sup_n x_n)$$

since g is monotone. So since Y is uniformly complete, there exists a unique map $h: \bar{\omega}^+ \rightarrow Y$ such that $h(n) = g(x_n)$, $h(\top) = g(f(\infty))$. And

$$h(\infty) = \sup_n g(x_n).$$

But gf is such a map; so by uniqueness, $h = gf$. So

$$g(\sup_n x_n) = g(f(\infty)) = h(\infty) = \sup_n g(x_n).$$

■

In particular, any map $f: \omega \rightarrow X$ extends uniquely to a map $\bar{f}: \bar{\omega} \rightarrow X$ and $\bar{f}(\infty) = \sup_n x_n$.

Proposition 12.5.3 *The uniformly complete modest sets form a small complete category.*

Proof. Let $X = \varprojlim X_i$ where the X_i are uniformly complete; we want to show that X is uniformly complete.

Suppose (x_n) is a chain in X ; then it gives rise to chains (x_{in}) in each X_i . To verify the first condition: since the X_i are uniformly complete, we get corresponding maps $\bar{\omega}^+ \rightarrow X_i$; these agree on each n ; they also agree on $\top$ (since the value at $\top$ is the value at ∞ , which is determined by the value at each n). So by uniqueness these maps form a cone; so we get a map $\bar{\omega}^+ \rightarrow X$ as desired. The second condition is verified in a similar way. ■

An example It might be instructive to think about what's going on in $\mathbf{R}$, the modest set corresponding to $\mathbf{R}$ itself. If $a \in \mathbf{R}$, we know that

$$a\{1, 2, 3, \dots\} = \bigcup_n a\{1, 2, \dots, n\};$$

$\mathbf{R}$ is link-ordered and chain complete, and furthermore every chain in $\mathbf{R}$ corresponds to a map $\omega \rightarrow \mathbf{R}$ (to prove this, observe first that all the equivalence classes are singletons). So the 'supremum map'

$$\mathbf{R}^\omega \rightarrow \mathbf{R}$$

is tracked by the element of $\mathbf{R}$ denoted by

$$\lambda x. x\{1, 2, 3, \dots\}.$$

Proposition 12.5.4 *Let $f: \mathbf{R} \rightarrow \mathbf{R}$ be any map (which must be tracked by an element $p \in \mathbf{R}$). Then*

$$\begin{array}{ccc} \mathbf{R}^\omega & \longrightarrow & \mathbf{R} \\ f^\omega \downarrow & & \downarrow f \\ \mathbf{R}^\omega & \longrightarrow & \mathbf{R}. \end{array}$$

Proof. The statement is again $\neg\neg$ -stable, so it follows immediately from continuity of application in the underlying model. ■

12.5.1 Least fixed points

If X is link-ordered and canonically focal, then the element $\perp \in X$ for which $\emptyset$ is a code is clearly the bottom element of X . And then we see, in the usual way, that

Proposition 12.5.5 *Let X be a chain complete, canonically focal link-ordered object. Then any map $X \rightarrow X$ has a least fixed point.*

But we might want to say a little more. Suppose $p \in \mathbf{R}$ tracks $f: X \rightarrow X$; then we get a map $\omega \rightarrow X$ tracked by

$$\{\langle \{0, \dots, n-1\}, k \rangle \mid k \in p^n \emptyset\}.$$

It would be nice if the least fixed point of f were coded by

$$yp = \bigcup_n p^n \emptyset$$

(recall that $y = \lambda f.(\lambda x.f(xx))(\lambda x.f(xx))$ denotes the *least* fixed point combinator in any graph model: see [8]). That is, if we could use a single fixed point combinator to compute fixed points in any domain.

This would follow if the extension $\bar{\omega} \rightarrow X$ were also tracked by f ; but *a priori* there's no reason why we shouldn't have to change codes in passing to the extension. This problem, and the way to get round it, is discussed further in section 13.3; the method described there is quite a general one.

12.6 Combining them

We know that the category of uniformly complete modest sets is quite well-behaved. All we need to verify now is that it actually contains the objects we would like it to. It certainly contains $\mathbf{N}$, since the order on that is discrete; and because it's complete, it's closed under the formation of total function spaces (as well as many other things). To show that it's closed under forming partial function spaces—spaces of computable partial maps—we need:

Conjecture 12.6.1 *If X is a uniformly complete modest set, so is $X_\perp$. (And so, in fact, $X_\perp$ is a uniformly complete focal modest set.)*

As an (external) statement about ordinary PERs on $\mathbf{R}$, this seems as if it ought to be immediate; but unfortunately I haven't been able to find a constructively valid proof of the corresponding statement about PERs on the object $\mathbf{R} \in \mathcal{E}$, i.e. a proof valid in $\mathcal{E}$. In fact it is not even obvious that the conjecture should be true. So we need to take a different approach.

In fact, all we want is a small complete category, all of whose objects are uniformly complete modest sets, containing $\mathbf{N}$ and closed under the lift construction. The obvious candidate is the category of replete objects; since Σ is closed under countable joins (easy to check by direct calculation), $\mathbf{N}$ is replete. So we only have to check that replete objects are uniformly complete.

But the category of uniformly complete modest sets is a small complete category. So to show that the category of replete objects will serve as a category of

predomains—that is, to show that every replete modest set is complete—we only need to establish the following:

Proposition 12.6.2 *Σ is a uniformly complete modest set.*

The proof is by direct calculation from the definition of Σ .

[Perhaps this seems like cheating; we couldn't prove something about the category we're worried about so we moved to a smaller one (just as in the previous section). But it's better to think of things the other way round: The replete objects, by definition, 'morally' have all the good properties they could possibly have; so the problem is to establish what these good properties are. What we've shown in this section is that among them are the properties of being link-ordered and uniformly complete, which is enough to justify treating the replete objects as predomains.]

For this to be useful, we'd better know that $\mathbf{N}$ is replete. Recall the proof that $\mathbf{N}$ was sober, and hence replete, in *Eff*: 10.3.3. The main thing that the argument relied on was that singletons were Σ -subsets of $\mathbf{N}$; so it suffices to prove

Lemma 12.6.3 *In the realizability topos $\mathcal{E}$ over the model R ,*

$$\forall n. \{n\} \subset_{\Sigma} \mathbf{N}.$$

Proof. Recall that the underlying set of $\mathbf{N}$ is just the natural numbers, with

$$|n| = \{n\}.$$

The classifier $\mathbf{N} \rightarrow \Sigma$ of $\{n\}$ is tracked by

$$\{(\{n\}, 0)\}.$$

■

So we'll just let our predomains be the replete objects. They form a small complete category, as we'd like them to.

By 10.4.3 the replete objects are closed under lifting; so it also makes sense to define a domain to be a replete object that's canonically focal. Then our domains will form an exponential ideal—in fact, by 12.1.1, they will be closed under arbitrary indexed products.

Note that since we have chain-complete partial orders floating around, we can use the usual inverse limit method to solve domain equations, if we like. Also note that we could replace ‘canonically focal’ by just ‘focal’ without losing too much.

Note. Perhaps this seems like cheating; we couldn’t prove something about the category we’re worried about so we moved to a smaller one (just as in the previous section). But to repeat the point made in section 10.4, it’s better to think of things the other way round: The replete objects, by definition, ‘morally’ have all the good properties they could possibly have; so the problem is to establish what these good properties are. What we’ve shown in this section is that among them are the properties of being link-ordered and uniformly complete, which is enough to justify treating the replete objects as predomains.

Note. We could also have used the methods of the following chapter to find a (presumably different) category of domains in our topos. It is after all odd that we should use such different methods with two such closely related models. But then, the whole question of how to relate different realizability toposes—even the correct notion of morphism between realizability toposes—is wide open.

Chapter 13

The Böhm tree model

13.1 A criterion for reflectivity

We'll run into a problem similar to the one mentioned at the beginning of the last chapter: The small complete categories in $\mathcal{E}ff$ that we studied had abstract definitions, and were complete for quite abstract reasons; but if we describe a subcategory of **Mod** by reference to PERs, then to prove that it's complete may involve calculating with PERs. The following result will be a useful tool. Note that all the statements are meant to be interpreted in the internal language.

Proposition 13.1.1 *Let C_0 be an (internal) collection of PERs such that*

- *if $(R_i)_{i \in I}$ is any internal family of PERs in C_0 , then their intersection $\bigcap_{i \in I} R_i$ is also in C_0 ;*
- *if $a \in A$ and $R \in C_0$ then $a^{-1}R \in C_0$, where*

$$p(a^{-1}R)q \quad \longleftrightarrow \quad (ap)R(aq);$$

- *the full subcategory C of **Mod** with object-of-objects C_0 is closed under the formation of equalizers in **Mod**; that is, if $R, S \in C_0$ and $a, b \in A$ track maps $A/R \rightrightarrows A/S$ then the PER E is in C_0 , where*

$$pEq \quad \longleftrightarrow \quad (pRq \wedge (ap)S(bq)).$$

Then there is a reflection functor $\mathbf{Mod} \rightarrow C$.

Proof. (Sketch.) Let R be any PER in $\mathbf{Mod}_0$; we'll construct a PER $\tilde{R}$ that's its reflection into $\mathbf{C}$. In fact, let

$$\tilde{R} = \bigcap_{S \in \mathbf{C}_0, R \subseteq S} S$$

and note that the identity $i \in A$ tracks a map $r: A/R \rightarrow A/\tilde{R}$.

To see that this gives us the reflection, let $a \in A$ track a map $f: A/R \rightarrow A/T$ where $T \in \mathbf{C}_0$.

Then since $\tilde{R} \subseteq a^{-1}T$, this map f factors through $A/\tilde{R}$ (a tracks a map $A/a^{-1}T \rightarrow A/T$, hence $A/\tilde{R} \rightarrow A/T$).

To see that this factorization is unique, let a, a' track maps $A/\tilde{R} \rightarrow A/T$, and let $E \subseteq \tilde{R}$ be a PER representing their equalizer, as above.

Then $E \in \mathbf{C}_0$ and $R \subseteq E$ (because a and a' both track factorizations of f); so $\tilde{R} \subseteq E$, i.e. $E = \tilde{R}$. ■

13.2 Partial maps and lifting

We'll be dealing with the closed term model $\mathbf{B}$ of the λ -calculus in which terms having the same Böhm tree are identified. $\mathcal{E}$ will denote the corresponding realizability topos, and $\mathbf{B}$ the assembly constructed from $\mathbf{B}$ in the standard way, as described earlier; we can regard $\mathbf{B}$ as an object of $\mathcal{E}$. By the way, the results in this section also hold for the Böhm tree model (see [8]), and can probably be extended to some other models.

We'll sometimes talk about a combinator like k or s as if it were an element of the model; what we really mean is the corresponding λ -expression. For example, we'll write i instead of $\lambda x.x$. Also, if M is a term, we will write $\text{BT}(M)$ for the Böhm tree of M . Finally, the symbol Ω will stand for the term $\Omega = (\lambda x.xx)(\lambda x.xx)$ (or any other unsolvable term; all are identified in the model).

[This is an unfortunate clash of notation, since we previously used Ω to denote the subobject classifier of $\mathcal{E}$. But both usages are standard; and anyway, one can always tell from the context whether Ω is a term or an object.]

Let Σ be the $\mathbf{B}$ -valued assembly with underlying set $\{\top, \perp\}$ and

$$|\perp| = \Omega, \quad |\top| = i.$$

The idea is that Ω represents divergence and i flags convergence. We think of Σ as being an object of $\mathcal{E}$. In fact it can be defined using the internal logic: it is simply the subset $\{M \mid \phi(M)\}$ of $\mathbf{B}$, where $\phi(M)$ is the predicate

$$\neg(M = \Omega) \rightarrow (M = i).$$

ϕ is clearly $\neg\neg$ -closed, so Σ is in fact a modest set. The important fact is that Σ is actually a subset of Ω , the object of truth-values. This is true since $\Sigma \mapsto \nabla 2$, but we can see it in a different way:

Proposition 13.2.1 *Σ is isomorphic to the subset*

$$\Sigma_1 = \{p \mid \exists! M \in \mathbf{B}. (\neg p \leftrightarrow (M = \Omega)) \wedge (p \leftrightarrow (M = i))\}$$

of Ω .

Proof. There is a map $\Sigma_1 \rightarrow \Sigma$ taking $p \mapsto M$. We can also define a map $\Sigma \rightarrow \Omega$ taking $M \mapsto |M = i|$; this actually maps into Σ_1 (we must check that $\neg(M = i) \leftrightarrow (M = \Omega)$ and $(M = i) \leftrightarrow (M = i)$, which is easy). These two maps are mutually inverse. ■

It seems likely that Σ is a dominance, but the proof would be rather messy. In any case we only need the following, weaker, result.

Proposition 13.2.2 *Σ is a dominance relative to Sep .*

Proof. We need to prove that $X \subset_{\Sigma} X$ for any object X , and that a Σ -subset of a Σ -subset is a Σ -subset. The first statement is easily seen to be true; so we simply need to show that if $X \subset_{\Sigma} Y$ and $Y \subset_{\Sigma} Z$ then $X \subset_{\Sigma} Z$.

Suppose M tracks the classifying map of $X \subset_{\Sigma} Y$, and N tracks the classifying map of $Y \subset_{\Sigma} Z$; then the classifier of $X \subset_{\Sigma} Z$ is tracked by

$$\lambda x. Nx(Mx)$$

(and observe that for any term P , if NP is unsolvable then so is $NP(MP)$, no matter what MP is).

[Note that this proof is very ‘uniform’ in character; it can easily be modified to prove the indexed versions of the conditions, which are what we really have to verify.] ■

Now to describe the lift of a modest set. Intuitively, if R is a PER then the ‘lifted PER’ $R_\perp$ is defined by: $\Omega R_\perp \Omega$, and if MRM then $M_\bullet R_\perp M_\bullet$ where $M_\bullet = \lambda z.ziM$. (The motivation is that i is a ‘flag’ indicating convergence of a computation; note that a computation might converge and yield the answer Ω , and this is different from diverging.) But we need to be able to formalize this definition in the internal logic; and we want the PER $R_\perp$ to be $\neg\neg$ -closed. We need to be a bit more careful.

So if R is a PER, define a PER $R_\perp$ by

$$MR_\perp N \iff \neg(M = N = \Omega) \rightarrow (((Mf)R(Nf) \wedge Mk = Nk = i))$$

(where $f = \lambda xy.y$). Some fiddling with intuitionistic logic verifies that this gives us a $\neg\neg$ -closed relation (using the fact that R is $\neg\neg$ -closed), as we require. Note that if we’d tried to define $R_\perp$ by cases—that is, by a formula containing a disjunction—it would not have been $\neg\neg$ -closed.

Proposition 13.2.3 *If $Y \cong \mathbf{B}/R$, then $Y_\perp \cong \mathbf{B}/R_\perp$ (where $Y_\perp$ denotes the Σ -partial map classifier relative to Sep).*

Proof. First observe that $Y \subset_\Sigma \mathbf{B}/R_\perp$; its classifier $\mathbf{B}/R_\perp \rightarrow \Sigma$ is tracked by

$$\lambda x.xi.$$

Suppose $X' \subset_\Sigma X$, M tracks a map $X' \rightarrow Y$ and N tracks the classifier $X \rightarrow \Sigma$ of X' . Then the corresponding map $X \rightarrow \mathbf{B}/R_\perp$ is tracked by

$$\lambda x.(Nx)(\lambda z.z(Mx))$$

(and note again that if NP is unsolvable then applying NP to $\lambda z.z(MP)$ will give us something unsolvable).

The converse is easy. ■

Call a PER R **canonically focal** if $\Omega R \Omega$; call a modest set a **canonically focal set** if it’s the quotient of $\mathbf{B}$ by a canonically focal PER. It follows from the above discussion that lifts of modest sets may be taken to be canonically focal. And 12.1.1 can be used to prove:

Proposition 13.2.4 *The fibration of canonically focal sets is closed under indexed products (over Sep).*

A reminder: Since the class of canonically focal PERs isn’t closed under isomorphism, a statement like this has to be interpreted with care.

13.3 A kind of chain completeness

In contrast to the situation in $\mathcal{E}ff$ and in chapter 12, there's no obvious way to define a useful partial order here. We have to define 'chain completeness' in a more concrete way, looking at the PERs themselves.

We will constantly be using the fact that 'application is continuous with respect to the Böhm tree topology': that is, if M is any term and N_n, N are terms with

$$BT(N_1) \subseteq BT(N_2) \subseteq BT(N_3) \dots$$

and

$$BT(N) = \bigcup_n BT(N_n)$$

then

$$BT(MN) = \bigcup_n BT(MN_n).$$

(This is actually a special case of a more general theorem; see [8, chapter 19]). We will abuse notation and write

$$M(\bigcup_n N_n) = \bigcup_n MN_n.$$

Now, let c_n denote the n th Church numeral and define

$$\ulcorner n \urcorner = \lambda f. c_n f \Omega;$$

that is,

$$\ulcorner 0 \urcorner = \lambda f. \Omega, \ulcorner 1 \urcorner = \lambda f. f \Omega, \ulcorner 2 \urcorner = \lambda f. f(f \Omega), \dots$$

Observe that, in a sense, the terms $\ulcorner n \urcorner$ form a sequence of approximations to the fixed point combinator

$$y = \lambda f. (\lambda x. f(xx))(\lambda x. f(xx));$$

we say this because

$$BT(y) = \bigcup_n BT(\ulcorner n \urcorner).$$

Call a PER R **complete** if it satisfies the following condition: for all terms M and N ,

$$(\forall n. (M\ulcorner n \urcorner)R(N\ulcorner n \urcorner)) \longrightarrow (My)R(Ny).$$

And call a modest set **complete** if it's the quotient of $\mathbf{B}$ by a complete PER.

Proposition 13.3.1 *The category of complete modest sets is reflective.*

Proof. Closure under intersection is obvious; and it's straightforward to show, using the fact that application is continuous, that the other hypotheses of 13.1.1 also hold. ■

There is one particularly interesting example of a complete PER. Let ω be the $\mathbf{B}$ -valued assembly with underlying set $\{0, 1, 2, \dots\}$ and put

$$|n| = \{^{\ulcorner n} \urcorner\};$$

and let $\bar{\omega}$ be the $\mathbf{B}$ -valued assembly with underlying set $\{0, 1, 2, \dots, \infty\}$ and put

$$|n| = \{^{\ulcorner n} \urcorner\}, |\infty| = \{^{\ulcorner y} \urcorner\};$$

there is an obvious mono $\omega \mapsto \bar{\omega}$.

Note that any map from ω into a complete modest set extends uniquely to $\bar{\omega}$; in fact, the defining condition for complete PERs tells us that if M tracks a map $\omega \rightarrow X$ then it also tracks the extension $\bar{\omega} \rightarrow X$.

We have a continuity result which is easy but important: Any map between complete modest sets ‘preserves sups of chains’; more precisely,

Proposition 13.3.2 *Let X and Y be complete modest sets, and let $f: X \rightarrow Y$. If $g: \omega \rightarrow X$ extends (uniquely) to $h: \bar{\omega} \rightarrow X$, then the unique extension of $fg: \omega \rightarrow Y$ to a map $\bar{\omega} \rightarrow Y$ is just fh .*

Proof. By the uniqueness of extensions, it suffices to show that fh extends fg ; that is, that they agree on ω . But they do, because h extends g . ■

And we do have sups of enough chains to be able to calculate fixed points (which, after all, was the main reason why we wanted ‘chain completeness’).

Proposition 13.3.3 *Suppose X is complete and canonically focal; then any map $X \rightarrow X$ has a fixed point. In fact, there is a fixed point operator $X^X \rightarrow X$.*

Proof. Suppose we have a map $X \rightarrow X$ tracked by a term P . Define $\omega \rightarrow X$ to be the map tracked by the term

$$M = \lambda x.xP;$$

this extends to a map $\bar{\omega} \rightarrow X$.

By the definition of complete PER, M tracks the extended map $\bar{\omega} \rightarrow X$. But $My = yP$ is a code for a fixed point of the map $X \rightarrow X$ (indeed, a fixed point ‘on the nose’ for codes).

The internal version—that there exists a fixed point operator $X^X \rightarrow X$, tracked by y —follows from this argument by adding some λ ’s. ■

Note the oddly intensional nature of this proof (y actually computes a fixed point ‘on the nose’ for codes); it has a rather different flavour from the usual fixed-point theorem for domains. I have not tried to formulate an appropriate notion of fixed-point induction yet.

13.4 All maps are stable?

The actual results in this section are rather trivial. The main thing to notice is that we have to be careful talking about meets.

In the λ -calculus this is how things go: Given terms M and N , say that $M \uparrow N$ if there exists a term P such that

$$\text{BT}(M), \text{BT}(N) \subseteq \text{BT}(P);$$

then we define $M \sqcap N$ to be the term with Böhm tree

$$\text{BT}(M) \cap \text{BT}(N).$$

(Note that in $\mathbf{B}$, this defines $M \sqcap N$ uniquely.) But this doesn’t internalize; there is no ‘meet’ map $\sqcap: \mathbf{B} \times \mathbf{B} \rightarrow \mathbf{B}$ in $\mathcal{E}$:

Proposition 13.4.1 *There does not exist a term Q such that*

$$M \uparrow N \longrightarrow QMN \simeq M \sqcap N$$

where $\simeq$ denotes equality of Böhm trees (equality in $\mathbf{B}$).

Proof. This is an easy consequence of [8, theorem 14.4.12]: Observe that for any terms M and N , $\Omega \uparrow N$ and $M \uparrow \Omega$, and so

$$Q\Omega N \simeq QM\Omega \simeq \Omega;$$

it follows that $QMN \simeq \Omega$ for all M, N . ■

However, we do have a $\neg\neg$ -closed relation

$$\mathbf{Meet} \stackrel{\text{def}}{=} \{(M, N, P) \mid (M \uparrow N) \wedge (P = M \sqcap N)\} \rightarrow \mathbf{B} \times \mathbf{B} \times \mathbf{B},$$

and this will have to do. We will sometimes abuse notation and write

$$P = M \sqcap N$$

as an abbreviation for $\langle M, N, P \rangle \in \mathbf{Meet}$.

Call a PER R **meet-closed** if

$$\forall P = M \sqcap N, P' = M' \sqcap N'. (MRM' \wedge NRN') \rightarrow (PRP');$$

and call a modest set $X \cong \mathbf{B}/R$ **meet-closed** if R is meet-closed. Note that we can then define a certain $\neg\neg$ -closed relation

$$\mathbf{Meet}_X \rightarrow X \times X \times X,$$

namely the quotient of $\mathbf{Meet}$ by R (in the obvious sense). And given $x_1, x_2, x \in X$, we will write

$$x = x_1 \sqcap x_2$$

as an abbreviation for $\langle x_1, x_2, x \rangle \in \mathbf{Meet}_X$.

The following results say that all maps between meet-closed PERs are stable, and that the category of meet-closed PERs is well-behaved.

Proposition 13.4.2 *If X and Y are meet-closed modest sets and $f: X \rightarrow Y$, then for all $x_1, x_2, x \in X$,*

$$x = x_1 \sqcap x_2 \longrightarrow f(x) = f(x_1) \sqcap f(x_2).$$

Proof. Immediate from the corresponding result for the underlying model: [8, corollary 14.4.11]. ■

Proposition 13.4.3 *The category of meet-closed modest sets is reflective.*

Proof. We'll use 13.1.1. It's clear that the meet-closed PERs are closed under arbitrary intersection.

Suppose $Q \in \mathbf{B}$ and R is meet-closed; we must show that $Q^{-1}R$ is meet-closed. But if

$$M(Q^{-1}R)M', N(Q^{-1}R)N'$$

then by definition,

$$(QM)R(QM'), (QN)R(QN');$$

but if

$$P = M \sqcap N, P' = M' \sqcap N'$$

then

$$QP = QM \sqcap QN, QP' = QM' \sqcap QN'$$

and so $(QP)R(QP')$, i.e. $P(Q^{-1}R)P'$.

As for equalizers: Suppose R, S are meet-closed PERs and Q_1, Q_2 track maps $\mathbf{B}/R \rightrightarrows \mathbf{B}/S$, and define the PER E as in proposition 13.1.1. Given

$$MEM', NEN'$$

and

$$P = M \sqcap N, P' = M' \sqcap N',$$

a similar calculation shows that PEP' . ■

Unfortunately, I don't know much about the behaviour of the meet relation under various constructions on objects. If we restrict ourselves to canonically focal meet-closed PERs, then binary products, function spaces and products indexed by $\neg\neg$ -sheaves (which can be calculated simply by taking the intersection of the relevant family of PERs) all have the right meet relations on them. But I can't prove any more general results; so at the moment, the usefulness of this concept is rather doubtful.

13.5 Putting it all together

As in the previous chapter, we'll fit all these pieces together by looking at the category $\mathcal{Rep}$ of replete objects (with respect to Σ).

Σ is replete; to prove that $\mathbf{N}$ is replete, it suffices to show that singletons are Σ -subsets of $\mathbf{N}$.

Lemma 13.5.1 *For all $n \in \mathbb{N}$,*

$$\{n\} \subset_{\Sigma} \mathbb{N}.$$

Proof. This follows from the fact that any partial recursive function is λ -definable. ■

Note that things are actually a bit subtler than they look. For example, there is no term M such that for *all* terms P ,

$$MP \simeq \begin{cases} i & \text{if } P = n; \\ \Omega & \text{else} \end{cases}$$

(cf. [8, exercise 20.6.14]).

Anyway we can now let our domains be the replete objects. $\mathcal{Rep}$ is closed under lifting; and any replete object is (isomorphic to an object which is) both complete and meet-closed, since the categories of complete PERs and meet-closed PERs are both small complete categories containing Σ .

We have lost something though. The predicates that carve the objects-of-objects of these two categories out of $\mathbf{Mod}_0$ are both $\neg\neg$ -stable, so

Proposition 13.5.2 *The object of complete PERs and the object of meet-closed PERs are both $\neg\neg$ -closed subobjects of $\mathbf{Mod}_0$.*

(In fact, so is the object of canonically focal PERs.)

That is, because of the ‘canonical’ nature of the sups of chains, meets and bottoms that are required to exist, the underlying objects of these categories are $\neg\neg$ -sheaves. In particular, we can calculate products indexed by these objects simply by taking the intersection of the given collection of PERs. The same is not necessarily true for the object of replete PERs (though it can be *made* true: see section 3.3).

Another problem: There’s no partial order floating around, so the inverse limit method doesn’t make sense; but we can still solve covariant domain equations by using completeness to show that initial algebras for functors exist.

Since there is a partial order on the underlying model—the specialization order arising from the tree topology—it should be possible to define a sensible partial order on certain PERs. But I just don’t know how to proceed. For example, you wouldn’t expect the partial order on a function space to be the pointwise order;

it should be something like the stable order. But you might expect the order on a product indexed by a $\neg\neg$ -sheaf to be pointwise. And how do we tie this up with the notion of meet-closure introduced in the previous section?

Finally: We have really glossed over the problem that the categories of complete PERs and meet-closed PERs are not closed under isomorphism, making their relationship with the category of replete objects more subtle than was the case for the categories of predomains discussed in previous chapters.

13.5.1 Closure under lifting

To avoid talking about the replete objects, we would have to show that the category of complete, meet-closed PERs is itself closed under the lift construction. Or rather—we have to be careful as we are dealing with a category which isn't closed under isomorphism—it is closed under the particular lift construction described explicitly in section 13.2. We couldn't obtain an analogous result in the previous chapter; but here it's possible, because of the special form of the definitions.

First we need a lemma that says we are allowed to prove these facts 'externally', and another that states the—external—continuity of application 'the other way round' (there's an internal counterpart, of course, but we won't need it).

Lemma 13.5.3 *Let $\mathcal{C}$ be a class of PERs, corresponding to a $\neg\neg$ -closed predicate ϕ on $\mathbf{Mod}_0$ (the internal object of $\neg\neg$ -closed PERs, which is a $\neg\neg$ -sheaf). In order to prove that*

$$\forall R \in \mathbf{Mod}_0. \phi(R)$$

is valid, it suffices to prove the external fact that $R \in \mathcal{C}$ for all PERs R .

Proof. Since $\mathbf{Mod}_0$ is a $\neg\neg$ -sheaf, the formula will be valid if and only if

$$\bigcap_R |\phi(R)| \neq \emptyset$$

(this follows from the description of the internal logic in terms of the tripos, and a little unwinding of the definitions). But since ϕ is the $\neg\neg$ -closed predicate corresponding to $\mathcal{C}$,

$$\phi(R) = \begin{cases} \mathbf{B} & \text{if } R \in \mathcal{C}; \\ \emptyset & \text{otherwise.} \end{cases}$$

So the intersection is nonempty precisely when $R \in \mathcal{C}$ for all R . ■

Lemma 13.5.4 *If $M = \bigcup_n M_n$ then $MN = \bigcup_n M_n N$.*

Proof. Define $a = \lambda x f.f x$. Then

$$MN = aNM = aN(\bigcup_n M_n) = \bigcup_n aNM_n = \bigcup_n M_n N$$

by continuity of application. ■

Proposition 13.5.5 *The category of complete PERs is closed under lifting.*

Proof. Let $\mathcal{C}$ be the collection of PERs R such that

if R is complete, so is $R_\perp$

(where $R_\perp$ is the PER defined in section 13.2; we are taking the external analogues of all the internal definitions). It corresponds to a certain $\neg\neg$ -closed predicate ϕ of the form

$$R \text{ complete} \longrightarrow R_\perp \text{ complete.}$$

The proposition is equivalent to the validity of

$$\forall R \in \mathbf{Mod}_0. \phi(R)$$

and so, by the first lemma, it suffices to prove the external fact that all PERs R on B belong to $\mathcal{C}$; that is: whenever R is a complete PER, then so is $R_\perp$.

First some preliminary definitions. For each $i \in \mathbb{N}$ let

$$s_i = \lambda f.f^i(\ulcorner n \urcorner f);$$

then $s_i \ulcorner n \urcorner = \ulcorner n + i \urcorner$ for all n . Now if M is any term let

$$M^{(i)} = \lambda x.M(s_i x);$$

then $M^{(i)} \ulcorner n \urcorner = M \ulcorner n + i \urcorner$ for all n . Observe that

$$My = \bigcup_n M \ulcorner n \urcorner = \bigcup_n M \ulcorner n + i \urcorner = \bigcup_n M^{(i)} \ulcorner n \urcorner = M^{(i)} y.$$

Now: Let R be a complete PER and suppose we have terms M, N such that for all n , $(M \ulcorner n \urcorner) R_\perp (N \ulcorner n \urcorner)$; we want to show that $(My) R_\perp (Ny)$. By the definition of $R_\perp$, for all n we have that if $\neg(M \ulcorner n \urcorner = N \ulcorner n \urcorner = \Omega)$ then

$$(M \ulcorner n \urcorner f) R (N \ulcorner n \urcorner f) \wedge M \ulcorner n \urcorner k = N \ulcorner n \urcorner k = i.$$

Suppose that $\neg(My = Ny = \Omega)$. Since $My = \bigcup_n M^n$ and $Ny = \bigcup_n N^n$, we must have

$$\neg(M^i = N^i = \Omega)$$

for some i sufficiently large. Then for all n , we have

$$\neg(M^{(i)}_n = N^{(i)}_n = \Omega)$$

and hence

$$(M^{(i)}_n)R(N^{(i)}_n)f;$$

this shows that $(M^{(i)}_n)R(N^{(i)}_n)f$ and hence $(My)R(Ny)$. Also,

$$M^{(i)}_n k = N^{(i)}_n k = i,$$

and hence

$$Myk = M(\bigcup_n M^n)k = (\bigcup_n M^n)k = \bigcup_n M^n k = \bigcup_n M^{(i)}_n k = \bigcup_n i = i;$$

and so $Myk = i$. Similarly for N . So we are done. ■

Proposition 13.5.6 *The category of meet-closed PERs is closed under lifting.*

Proof. Similar, but easier; instead of continuity of application, we use stability of application. The details are left to the reader. ■

We can combine the proofs of these two propositions to show that the category of complete, meet-closed PERs is closed under lifting—which is what we wanted.

Note. In view of some of the arguments we've used, it seems as if it might be more natural to use Nakajima trees rather than Böhm trees (see [8, exercise 19.4.4]); that is, it might come in handy to have the η -rule around. But then there would be a lot of theorems to reprove.

Chapter 14

Related work and loose ends

14.1 Fixed point categories

As mentioned in the historical sketch in 1.3, this was an idea of Dana Scott's. It seems to be the first attempt at an axiomatic (or synthetic) approach to domain theory based on simple categorical principles. We saw the definition in 7.1, but let me repeat it.

Let $\mathcal{C}$ be a cartesian closed category. Following Lawvere, we say that an object $X \in \mathcal{C}$ has the **fixed point property** if, for any object $Y \in \mathcal{C}$, any map $f: X^Y \rightarrow X^Y$ has a fixed point. The **fixed point category** of $\mathcal{C}$, denoted $\text{fix}\mathcal{C}$, is the full subcategory of $\mathcal{C}$ whose objects are the objects with the fixed point property.

[Note that—if $\mathcal{C}$ is a topos, for example—the fixed point property is not a property definable in the internal language, since it involves quantifying over objects of $\mathcal{C}$. That's one nasty feature of this definition.]

Clearly $\text{fix}\mathcal{C}$ is an exponential ideal of $\mathcal{C}$. Scott proved, using a trick, that $\text{fix}\mathcal{C}$ is also closed under binary products. And of course it contains the terminal object. We could throw away some of this information and simply say that

Proposition 14.1.1 (Scott) *$\text{fix}\mathcal{C}$ is a cartesian closed category.*

The idea was to find toposes with nontrivial fixed point categories. As we mentioned in 7.1, Rosolini and Hyland established some time ago that $\mathcal{E}ff$ had quite a big fixed point category, by showing (via direct calculations) that Σ had the fixed point property; but after that, no more work was done along those lines.

One other problem. Suppose that we are interested in modelling dependent types, so we have a category $\mathcal{C}$ which is locally cartesian closed (that is, it has

pullbacks and every fibre is cartesian closed). Each fibre has a fixed point category; do we actually have a *subfibration* of $\text{cod}: \mathcal{C}^2 \rightarrow \mathcal{C}$? Even if $\mathcal{C}$ is a topos this is not obvious; as mentioned above, we aren't dealing with a property expressible in the internal language which would automatically be preserved by pullback functors.

In fact I don't see why it should be true in general (a proof by direct calculation goes smoothly up to a certain point; then the snag is that once you pull back a family of objects, you may get new families of endomorphisms that you didn't have before, because they're not constant on the fibres of the reindexing map). I don't think this was something that Scott was worried about; in any case, it may be possible to tweak the definition so that things work out.

It's fair to say, though, that at the moment the notion of fixed point category seems like a bit of a red herring. As pointed out in 7.1, the fixed point category in $\mathcal{E}ff$ is far too big; this would seem to suggest that, though the *property* is of interest, the *category* is not. [The same could be said about complete Σ -spaces! But at least they have better closure properties.]

14.2 The work of Rosolini

Here I just want to make a few remarks about the work that appears in Rosolini's thesis [68], and in [69]. We'll think of everything as going on inside $\mathcal{E}ff$, though that's not the point of view that he took.

Maybe the whole problem with section 10.3 was that we took the word 'topology' too literally. Dana Scott suggested that one should try and work with a looser definition of topology—one in which open sets need only be closed under *countable* joins—which is better adapted to Σ . For example, now the Σ -subsets of any object form, on their own, a topology on that object in this new sense. (But the problem of 10.5.3 remains.)

This was the point of view taken by Rosolini; the theory he develops is based on the 'sober' spaces for this new notion of topology. Namely: after Scott, let $[[X]] \mapsto \Sigma^{\Sigma^X}$ denote the object of maps $\Sigma^X \rightarrow \Sigma$ preserving binary meets and countable joins; and call X a σ -set if the natural inclusion $X \mapsto [[X]]$ is an isomorphism.

It is easy to see that any σ -set is (in our terminology) a complete Σ -space; so it makes sense to think of them as predomains. There are plenty of examples of σ -sets, and they have some closure properties—though we still don't know if the

category of σ -sets has all limits, or even if it's cartesian closed.

At this point the theory takes a more classical turn with the introduction of countable bases. Call $b \in X$ **basic** if

$$\{x \in X \mid b \leq x\} \subseteq_{\Sigma} X;$$

and call X a **σ -domain** if the subobject B of basic elements is bounded complete and countable, with a decidable order, and if every element of X is the sup of a chain in B . Then

Proposition 14.2.1 (Rosolini) *The σ -domains are closed under finite products, $\mathbb{N}$ -indexed coproducts, partial function spaces and limits of projections.*

Since 1 and $\mathbb{N}$ are σ -domains, this gives us a pretty large collection of domains. In fact, it seems to be the first category of domains in $\mathcal{E}ff$ whose definition was completely internal. (Since then, Scott (in [74]) has proposed an internal definition of ‘SFP-domain’ in $\mathcal{E}ff$; and probably other variations are possible.)

There are two objections. Firstly, these categories just don't seem to have enough closure properties. Secondly, everything is starting to look just like conventional domain theory with all its complications. This is not just a practical point—that the mathematics is no longer easy—it's a sort of philosophical problem as well: namely, why do we want to bother with countable bases in what's meant to be a clean, synthetic theory where computability is already built in?

(One could argue that it's nice to have some formal analogue of the notion of finite approximation. But it's not *that* nice; and also, if one wants to talk about such things, it seems more appropriate to do so in a more concrete setting—either classical domain theory or some other specific model, such as a graph model or the continuous models studied by Amadio and Abadi/Plotkin. That is, it might be worth analysing such things, but that point of view doesn't fit in very well with the goal of developing a high-level theory with minimal fuss: the ‘domain theory toolkit’ ideal.)

It should be emphasized, though, that Rosolini's thesis was a remarkable landmark. Besides containing fundamental work on categories of partial maps (which we have been using continually in this dissertation), and many other results, it must be regarded as the first place in which synthetic domain theory was systematically and convincingly developed.

14.3 Extensional PERs

The research leading to the paper [21] (by Freyd, Mulry, Rosolini and Scott), and the work in this thesis on domains in $\mathcal{E}ff$, were carried out independently at about the same time (early 1989). Some of the basic results are remarkably similar; so its interesting how different the two approaches are. The authors of [21] adopted a very concrete point of view, preferring explicit calculations with PERs (even more hands-on than chapters 12 and 13) to more abstract, high-level methods; and the fact that everything is going on inside an ambient topos is not really exploited.

The definitions in their paper are, on the face of it, very external; but I believe they all make sense internally. (We mention an alternative, genuinely internal, approach below.) Let S be any set of natural numbers; say that a PER R is **extensional mod S** if for all $e \in \mathbf{N}$ such that eRe and for all $f \in \mathbf{N}$,

$$eRf \iff \forall n \in S. (\{e\}(n) \downarrow \leftrightarrow \{f\}(n) \downarrow) \wedge (\{e\}(n) \downarrow \rightarrow \{e\}(n) = \{f\}(n)).$$

Call R *extensional* if there exists some S such that R is extensional mod S (note that if R is extensional then there is a unique choice of S). The PERs we are interested in are then the extensional PERs whose quotients are complete Σ -spaces.

One can go on to show, essentially by brute force, that the corresponding full subcategory of Mod is cartesian closed and in fact reflective, as well as being very well-behaved with respect to strict maps and lifting. It's also possible to prove a result about polymorphic parametricity (the corresponding question for complete Σ -spaces remains open).

The concrete form of the definition is a little inconvenient, because we don't get a category closed under isomorphism, and because the proofs all involve a lot of stuffing around with Turing machines (the same sort of thing as in section 7.3). However, it is immediate from the definition that the object-of-objects of this category is a $\neg\neg$ -sheaf, which was one of the reasons why it was cast in this form (though this is also true of the alternative definition mentioned below).

Incidentally, not every complete Σ -space is (isomorphic to) an extensional PER; a counterexample was found by Hyland. Roughly speaking, start with an extensional PER whose quotient is the chain classifier ω defined in section 6.1; pick a number from each sequence of integers coding a non-stationary chain in ω , and remove this number from the domain of the PER. What you get is something

with the same global sections as ω , but complete (since every chain is eventually stationary). Of course, there are details to be filled in here.

As Hyland points out—and apparently, this idea was close to the original approach taken by the authors of [21]—a more internal definition is possible. We could call a Σ -space X **extensional** if

$$X \mapsto \Sigma^{\Sigma^X}$$

were $\neg\neg$ -closed; the existence of a reflection then follows from the (epi)-(regular mono) factorization system in *Sep* (just as the reflection into Σ -spaces boils down to the (regular epi)-(mono) factorization system). The idea doesn't seem to have been further developed in print—though see [71].

It's perhaps worth pointing out that Hyland has found an example of an extensional PER which is not a replete object (cf. 10.3.4). So it seems as if the category of extensional PERs is itself too big.

One advantage of extensional PERs is that they seem more amenable to direct calculations than, say, complete Σ -spaces or replete objects. For example, Freyd has shown in this way that if T is a functor on extensional PERs, the initial algebra and final coalgebra for T are isomorphic. But there ought to be a proof based on much more abstract considerations, applicable to the replete objects at least.

14.4 Domains in other models

Quite a lot of people, from McCarty ([46]) onwards, have been looking for domains in $\mathcal{E}ff$. Not many have been working with other combinatory algebras: Amadio (see for example [3]), and Abadi and Plotkin (see [1]).

Again, these authors work entirely in the context of the PER model as sketched in section 2.1; in fact it's not even clear what their results might mean in the internal language of the corresponding realizability topos. In this section I will just give a quick summary of [1], to give an indication of the current state of knowledge. Everything seems to work pretty differently from the way things work in $\mathcal{E}ff$.

Start with a certain kind of domain model of the λ -calculus; that is, a cpo D with arbitrary meets together with (roughly speaking)

- projections $\pi_n: D \rightarrow D$ with finite range such that $\pi_0 = \perp$ and $\sup_n \pi_n = 1$;

- retractions from D into $\mathbf{O}$, $\mathbf{N}$ $D \times D$, D^D , all compatible with the π_n .

where $\mathbf{O}$ is the two-point cpo (which is going to play a role similar to that of Σ). All this structure must be compatible with meets.

Now call a PER S

complete if $\perp X \perp$ and if, whenever $R \subset S$ is directed (as a subset of D^2) then

$$(\bigsqcup R)(\bigsqcup R);$$

uniform if xSy implies $\pi_n x S \pi_n y$ for all n ;

antisymmetric if the Σ -preorder on D/S (defined as in 5.2 using $\Sigma = \mathbf{O}$) is a partial order;

meet-closed if whenever we have families (x_λ) , (y_λ) such that $x_\lambda S y_\lambda$, then

$$(\prod_\lambda x_\lambda) S (\prod_\lambda y_\lambda);$$

convex if whenever $x \sqsubseteq y \sqsubseteq z$ and xSz then xSy

where $\sqsubseteq$ denotes the partial order on D . (The meet-closure condition is actually optional.)

The category of PERs with all these properties is cartesian closed; it is an $\mathbf{O}$ -category, so we can solve domain equations by the inverse limit method; and it is closed under intersection, giving us a model for the second order typed λ -calculus—and probably the calculus of constructions, though the details are not written out there (this would be a bit more complicated to verify).

There's actually somewhat more to the paper than this; we've just listed the basic definitions, so that they can be compared with the ones in chapter 13. See also [4] for more information on current results available about continuous models.

On internal versus external methods

It seems worth repeating a remark which applies both to the work of Abadi/Plotkin and Amadio just referred to, and to the work on extensional PERs mentioned before. Namely: rather than using the internal language of an appropriate realizability topos, they work in an external manner with PERs.

One has to be very careful in doing this. For example, it's not safe to regard a Σ -space with its Σ -order simply as an ordinary set of equivalence classes that happen to be partially ordered in a canonical way; that is, as a poset in the ordinary category of sets. (E.g. both Σ and the object Z_A of 5.4.2 give us the same two-point poset, which is of course chain-complete, but they are quite different objects—the fact that the classical poset that Z_A gives rise to has sups of chains is simply irrelevant.) Similarly it's not appropriate to talk about 'all' functors from the ordinary category $\mathbf{Per}$ into itself.

The solution these authors adopt is to give explicitly effective versions of their definitions: effectively given chains, realizable functors and so on. But if we work in the internal language of a realizability topos, there's no need to worry! All the chains are effectively given, all the functors are realizable; these things just correspond to the fact that they are internally definable. So we can really avoid carrying around all these codes.

Another advantage of working internally is that the 'indexed' versions of results come for free. (We need them, for example, to interpret the theory of constructions.) Of course, there are external proofs—'uniformly parametrized' versions of the previous external proofs—but this is an additional complication.

Finally, the fact that you're working in an ambient topos means that there is an ambient set-theoretic logic available; hopefully this will be helpful in putting together appropriate program logics. Certainly the set-theoretic point of view has an attractive simplicity even apart from this sort of practical consideration.

Well, this is rather glib. The fact is that it's *not* yet clear—at least in the case of the work on continuous models mentioned in this section—what the proper internal versions of the definitions should be. But I'm confident that they'll emerge eventually, and that topos theory will turn out to be the more convenient way to do things, just as it was for domains in *Eff*.

14.4.1 Domains in Grothendieck toposes

As I mentioned earlier, Scott showed how to embed categories of domains in presheaf categories; and Rosolini had already studied certain Grothendieck toposes in his thesis [68]. But since then, attention seems to have been focussed on realizability toposes (equivalently, PER models). Only recently has attention started to turn back to such toposes, which can contain various classical categories of

domains as full subcategories.

Work is being done in this direction by Paul Taylor (see [77]) and particularly by Gareth Badenhorst, a student of Hyland's, and Jørn Schmidt. Badenhorst has suggested that certain gros toposes might lead to models of polymorphism. However, this is still very much work in progress.

14.5 Synthetic domain theory

It's hard to say what does and doesn't fit under this heading. I just want to mention two examples that illustrate different ways of proceeding: namely, some ideas that Peter Freyd hints at in [21] and [17]; and a weird but promising-looking proposal being developed by Martin Hyland.

It's worth pointing out that the two approaches can't really be compared; they have rather different motivations, and possibly different aims. Here we will just try and indicate the flavour of the approaches rather than giving a full exposition.

14.5.1 The approach of Freyd

This is more a series of suggestions than a fully developed theory. See [17] and [20].

The first idea is to think of a category of domains as being a certain sort of category enriched over $\omega\text{-Ord}_1$, the category of chain-complete posets with a bottom element and strict maps (relative to some base topos). Freyd imposes some additional axioms that suffice to show that solving mixed-variance domain equations can be reduced to solving covariant domain equations: [17].

The second idea is based on Freyd's so-called *principle of versality*: a data type should be characterized by both a universal and a co-universal property. He defines an *algebraically compact category* to be one in which every functor has both an initial algebra and a final coalgebra, which are canonically isomorphic. The category of extensional PERs is shown to be algebraically compact. A connection with the fixed point property appears.

14.5.2 The approach of Hyland

These proposals are at a somewhat more advanced stage of development: see [30]. The central idea here is the notion of *replete* object relative to a dominance, as defined in 10.3. There are several groups of fairly simple axioms.

We won't repeat them here. The general features are:

- There is a dominance Σ ; decidable subsets are Σ -subsets; countable unions of Σ -subsets are Σ -subsets;
- Lifting and co-lifting form strong monads with certain properties.
- The intrinsic order on Σ is characterized by maps $\Sigma \rightarrow \Sigma$ (cf. the definition of linked Σ -space).
- The map $\omega \rightarrow \bar{\omega}$ is Σ -anodyne.

The set of axioms is now reasonably stable, and one can prove most of the things one might hope to prove.

The point is to have more 'algebraic' foundations for domain theory, less sensitive to particular logical principles that hold in a given topos. For example, the theory of Σ -spaces relies heavily on the fact that Markov's principle holds in $\mathcal{E}\mathcal{F}$; one could say that domain theory shouldn't rely on a fact like this.

[But one has to make a distinction between two goals here: following up the point of view that domains are sets in some computational universe is *not* the same thing as looking for axioms for domain theory that don't involve an ambient topos, that is, that don't reduce domain theory to set theory. In the first case Markov's principle may express something quite sensible about the computations that can go on, whereas in the second case it's quite artificial.]

In [77], Taylor proposes some axioms for a category of domains also based on the idea of replete object (discovered independently by him); they look somewhat more conventional than Hyland's. He uses them to give an analysis of the fixed point property.

Finally, I should mention the work of Crole and Pitts which appears in [15]. Beginning with the idea of 'computation types' described in [47], they develop a type theory containing an object rather like the object $\bar{\omega}$ defined in section 6.1; in this (entirely syntactic) framework, a lot of denotational semantics can be done. Both ordinary categories of domains and complete Σ -spaces form models for this theory.

One could argue that in the end, it's syntactic theories like this that people are going to use—not real categories of domains. Models should just be a source of

ideas and understanding for the theoreticians, and people like language designers shouldn't have to worry about them.

On the other hand, if domain theory can be presented in a sufficiently simple-minded 'set-theoretic' way, it's not unreasonable to expect non-mathematicians to be interested in using it. Certainly it would then be at least as accessible as type theory (which is presumably understood by at least some non-mathematicians).

14.6 Sound predicates

The following question arose in relation to 6.4.

Suppose $\mathcal{C}$ is a 'good' category of (pre)domains and X is an object of $\mathcal{C}$. For what kind of predicates ϕ will it be true that

$$\{x \in X \mid \phi(x)\}$$

will also be an object of $\mathcal{C}$? In other words, when can we use the internal language of the topos to carry out constructions on predomains?

For example, we constructed lax limits of linked Σ -spaces by carving out subobjects of certain products, and we relied on 5.4.6 to tell us that the resulting subobjects were linked; and then, in the case of complete Σ -spaces, a further calculation was required to show closure under sups of chains.

So it wasn't clear there that lax limits of *replete* Σ -spaces could be constructed in the same way. But they can. The reason is that the predicates we used were 'essentially algebraic' and were defined using the Σ -order; and $\mathcal{R}ep$ has finite limits and contains the object Σ , which classifies the Σ -order.

For instance: Suppose $X \in \mathcal{R}ep$ and $x_0 \in X$; consider the subobject

$$X' = \{x \in X \mid x_0 \leq x\}$$

of X ; we want to show that this is also in $\mathcal{R}ep$. But a little thought shows that there's an equalizer diagram

$$X' \rightarrow X^{\Sigma} \rightrightarrows X$$

where the two parallel arrows are evaluation at $\perp$ and the constant map at x_0 .

If we are trying to construct a formal language for domain theory—rather than just type theory—it might be useful to be able to define (sub)domains using certain predicates. But if the intended model is some category $\mathcal{C}$, we need to know

what classes of predicates are **sound** for $\mathcal{C}$, in the sense that $\mathcal{C}$ is closed under taking extensions of such predicates. If we're modelling dependent types, then the meanings of these predicates should also be stable under pullback.

Well, we know that we can talk about equality, the partial order, suprema of chains and Σ -subsets; and we're allowed to build up formulas from these basic predicates provided we stay in the finite $\lim$ fragment of first-order logic. In fact, we can probably get away with more. But I can't suggest any concrete ways of extending type theories to take all this into account.

Incidentally, these considerations should be closely related to the problem of finding an appropriate program logic for synthetic domain theory; this problem has barely been touched.

14.7 Final remarks

There must be a way to explain everything more simply. Remember that simplicity was one of our original goals; and contrast this with the mass of technical machinery that we seem to have used (reread chapter 2 if you like).

Actually, there is a general problem behind all this. Almost everything that looks strange or difficult—the discussion of completeness in section 2.6, the problems in making the proof of 9.1.3 rigorous, the strange ‘approximation by decidable truth-values’ argument that appears in several places (especially chapter 7)—almost all of it boils down to the fact that we must have definitions and proofs which make sense constructively. Constructive reasoning does not feel natural. Perhaps it should; but it doesn't, and the disadvantage of having to think in an unfamiliar way almost cancels out the advantage of working in a clean set-theoretic (topos-theoretic) framework.

In the long term, I think this problem will become less serious. As the importance of intuitionistic logic and topos theory becomes more widely appreciated, we will grow more used to reasoning constructively. After all, quite a few mathematicians have picked up the same habit; and it hasn't, in the end, caused them too much suffering.

The same argument applies to the use of powerful tools from category theory. At the moment, these may sometimes seem quite technical. Eventually they will be second nature; not an obstacle to understanding, but an aid. Perhaps this is already happening. (And besides, we didn't really use that much category theory.)

Bearing this in mind... The material in chapters 5 to 9 is in a pretty satisfactory state, but I'm sure that it's possible to present it in a more accessible way. This will have to be done eventually, if Σ -spaces are ever to be useful.

It will also be worthwhile to reexamine the various applications of domain theory, to see whether the same techniques work in the context of Σ -spaces; section 7.5 is an example, but there are many others. This activity is not just pragmatic; it may also shed some light on exactly what it is that our domain theory is analysing.

The chapter on sheaves doesn't really contain any compelling evidence that they're going to be interesting or useful. In fact, I think that we won't make much more progress till we understand the topological aspects of Σ -spaces a bit better; and that may take some time. But I also feel that the effort will turn out to be worthwhile.

It's disturbing that so many of the definitions in chapter 12, and especially chapter 13, are phrased in terms of PERs. There ought to be categorical ('extensional') analogues; this would make everything much cleaner. But perhaps one shouldn't expect an easy answer. After all, in *Eff* things are subtle; for instance, orthogonality to $\omega \mapsto \bar{\omega}$ doesn't tell the whole story about chain completeness (even if, in the end, we plan to forget about all the rest of it).

On a more general level, the actual aims of the theory have to be worked out more clearly. As I mentioned, different people seem to have had different goals in mind, and this may explain why they have adopted such different approaches. Some practical philosophy is called for; if all these points of view are not properly analysed and related, the situation will remain confused. I'm not suggesting that there is only one proper goal, any more than that there is one proper method; I just believe that the various meanings of 'synthetic domain theory' have to be explained and justified more clearly, and that this might turn out to be surprisingly difficult.

There are also plenty of more practical questions to think about. For example, what happens when we actually use these domains to do denotational semantics? (E.g. how do the powerdomains behave?) Is there a sensible treatment of subtyping? (PERs have certainly been used to model subtypes before, but in a way which seems hard to interpret categorically: see, most recently, [71].) What about a domain theory that provides a sensible treatment of functional I/O? (Models based on function realizability may turn out to be a good way of dealing with streams: which—if they represent input channels from the outside world—should

not be thought of as lazy lists.) And so on...

All in all, there seems to be a lot left to do. The only thing I'm sure of is that the final story, whatever it is, will look quite different from the fragments written down here.

Bibliography

- [1] M. Abadi, G. Plotkin: A PER model of polymorphism and recursive types, in: Proc. of 5th Annual Symposium on Logic in Computer Science, 1990.
- [2] S. Abramsky: Domain theory in logical form, in: Proc. of 2nd Annual Symposium on Logic in Computer Science, 1987.
- [3] R. Amadio: Recursion over realizability structures, to appear in *Information and Computation*.
- [4] R. Amadio: Domains in a realizability framework, LIENS-TR 19-90.
- [5] R. Amadio, L. Cardelli: *Subtyping recursive types*, DEC SRC Research Report 62, August 1990.
- [6] M. Artin, A. Grothendieck, J. L. Verdier: *Théorie des Topos et Cohomologie Étale des Schémas (SGA4 Vol. 1)*, Lecture Notes in Mathematics 269, Springer, 1972.
- [7] E. S. Bainbridge, P. J. Freyd, A. Scedrov, P. J. Scott: Functorial polymorphism: preliminary report, Proceedings of the Programming Institute on Logical Foundations of Functional Programmin (Austin, 1987).
- [8] H. P. Barendregt: *The Lambda-Calculus: Its Syntax and Semantics (revised edition)*, North-Holland, 1984.
- [9] M. Beeson: *Foundations of Constructive Mathematics*, Springer, 1985.
- [10] J. Benabou: Fibred categories and the foundations of naive category theory, *J. Symbolic Logic*, vol. 50, no. 1 (1985) 10-37.
- [11] D. Bridges, F. Richman: *Varieties of Constructive Mathematics*, Cambridge University Press, 1987.

- [12] A. Carboni, P. Freyd, A. Scedrov: A categorical approach to realizability and polymorphic types, in: *Lecture Notes in Computer Science* 298, Springer, 1988.
- [13] L. Cardelli, G. Longo: *A semantic basis for Quest*, DEC SRC Research Report 55, February 1990.
- [14] T. Coquand, G. Huet: Constructions: a higher-order proof system for mechanizing mathematics, in: *EUROCAL '85: European Conference on Computer Algebra*, Springer, 1985.
- [15] R. Crole, A. M. Pitts: New foundations for fixpoint computations, in: *Proc. of 5th Annual Symposium on Logic in Computer Science*, 1990.
- [16] K. E. Flannery, J. J. Martin: The Hoare and Smyth power domain constructors commute under composition, *Journal of Computer and System Sciences* **40**, 125–135 (1990).
- [17] P. Freyd: Recursive types reduced to inductive types, in: *Proc. of 5th Annual Symposium on Logic in Computer Science*, 1990.
- [18] P. J. Freyd: POLYNAT in Per, in *Categories in Computer Science and Logic (Proc. Boulder 1987)*, *Contemp. Math.* **92** (1989).
- [19] P. J. Freyd: Structural polymorphism, to appear.
- [20] P. J. Freyd: Algebraically complete categories, to appear in the proceedings of *ICTM '90* (Como).
- [21] P. Freyd, P. Mulry, G. Rosolini, D.S. Scott: Extensional PERs, in: *Proc. of 5th Annual Symposium on Logic in Computer Science*, 1990.
- [22] J.-Y. Girard: *Interprétation fonctionnelle et élimination des coupures de l'arithmétique d'ordre supérieur*, Thèse de Doctorat d'État, Paris, 1972.
- [23] J.-Y. Girard, Y. Lafont, P. Taylor: *Proofs and Types*, Cambridge University Press, 1989.
- [24] C. A. Gunter: Relating total and partial correctness interpretations of non-deterministic programs, in: *17th Annual ACM Symposium on Principles of Programming Languages*, ACM Press, 1990.

- [25] C. A. Gunter, D.S. Scott: Semantic domains, preprint, 1986.
- [26] J. Hughes, P. Wadler: Projections for strictness analysis, in: *IFIP Conference on Functional Programming Languages and Computer Architecture*, Lecture Notes in Computer Science 274, Springer, 1987.
- [27] J. M. E. Hyland: The effective topos, in: *The L. E. J. Brouwer Centenary Symposium* (ed. A. S. Troelstra, D. van Dalen), North-Holland, 1982.
- [28] J. M. E. Hyland: A small complete category, *Annals of Pure and Applied Logic* 40 (1988) 135-165.
- [29] J. M. E. Hyland: Synthetic domains—the story so far, *Abstracts EACTS*, 1990.
- [30] J. M. E. Hyland: First steps in synthetic domain theory, to appear in the proceedings of *ICTM '90* (Como).
- [31] J. M. E. Hyland, P. T. Johnstone, A. M. Pitts: Tripos theory, *Math. Proc. Camb. Phil. Soc.* (1980), **88**, 205.
- [32] J. M. E. Hyland, A. M. Pitts: The theory of constructions, categorical semantics and topos-theoretic models, *Categories in Computer Science and Logic (Proc. Boulder 1987)*, *Contemp. Math.* **92** (1989) 137-199.
- [33] J. M. E. Hyland, E. P. Robinson, G. Rosolini: The discrete objects in the effective topos, *Proc. Lond. Math. Soc.* (3) 60 (1990) 1-36.
- [34] J. M. E. Hyland, E. P. Robinson, G. Rosolini: Algebraic types in PER models, preprint.
- [35] P. T. Johnstone: *Topos Theory*, Academic Press, 1977.
- [36] P. T. Johnstone: Scott is not always sober, in: *Lecture Notes in Mathematics* 871, Springer, 1981.
- [37] P. T. Johnstone: Open maps of toposes, *manuscripta math.* 31 (1980) 217-247.
- [38] P. T. Johnstone, I. Moerdijk: Local maps of toposes, *Proc. Lond. Math. Soc.* (3) 58 (1989) 281-305.

- [39] P. T. Johnstone, E. P. Robinson: A note on inequivalence of realizability toposes, *Math. Proc. Camb. Phil. Soc.* (1989), **105**, 1.
- [40] A. Joyal, M. Tierney: *An Extension of the Galois Theory of Grothendieck*, Memoirs AMS No. 309, 1984.
- [41] G. M. Kelly: *Basic Concepts of Enriched Category Theory*, Cambridge University Press, 1982.
- [42] A. Kock, J. Schmidt: A coherent theory of sites, *Bull. Soc. Math. Belgique (Série A)*, Tome XLI Fasc. 2 (1989) 321-331.
- [43] G. Kreisel: Interpretation of analysis by means of constructive functionals of finite type, in: *Constructivity in Mathematics* (ed. A. Heyting), North-Holland, 1959.
- [44] J. Launchbury: Dependent sums express separation of binding times, 1990.
- [45] G. Longo, E. Moggi: Constructive natural deduction and its ' ω -set' interpretation, *Math. Struct. in Comp. Science* (1991), vol. 1.
- [46] D. C. McCarty: *Realizability and Recursive Mathematics*, D. Phil. thesis (Oxford; Carnegie-Mellon), 1984.
- [47] E. Moggi: Computational lambda-calculus and monads, preprint, University of Edinburgh, 1988.
- [48] J. van Oosten: Extension of Lifschitz' realisability to higher order arithmetic, and a solution to a problem of F. Richman, preprint, 1990.
- [49] V.C.V. de Paiva: *Subtyping in Ponder (preliminary report)*, University of Cambridge Computer Laboratory Technical Report No. 203, August 1990.
- [50] W. K.-S. Phoa: Relative computability in the effective topos, *Math. Proc. Camb. Phil. Soc.* (1989), **106**, 419.
- [51] W. K.-S. Phoa: Categorical foundations of effective domain theory, preprint.
- [52] W. K.-S. Phoa: *The synthetic theory of effective domains: a categorical and topos-theoretic study*, J.T. Knight Prize essay, 1990.

- [53] W. K.-S. Phoa: Effective domains and intrinsic structure, in: Proc. of 5th Annual Symposium on Logic in Computer Science, 1990.
- [54] W. K.-S. Phoa: Building domains from models of computation, submitted.
- [55] W. K.-S. Phoa: Reflectivity for categories of synthetic domains, submitted, 1990.
- [56] W. K.-S. Phoa: From term models to domains, to appear in the proceedings of Theoretical Aspects of Computer Software (Sendai, 1991).
- [57] W. K.-S. Phoa: Two results on set-theoretic polymorphism, to appear in the proceedings of Category Theory and Computer Science (Paris, 1991).
- [58] W. K.-S. Phoa: Using fibrations to understand subtypes, preprint.
- [59] W. K.-S. Phoa, P. Taylor: The synthetic Plotkin powerdomain, preprint.
- [60] D. H. Pitt, A. Poigné, D. E. Rydeheard: *Proc. Category Theory and Computer Science*, Lecture Notes in Computer Science 283, Springer, 1987.
- [61] A. M. Pitts: *The Theory of Triples*, Ph.D. thesis (Cambridge), 1981.
- [62] G. D. Plotkin, M. B. Smyth: The category-theoretic solution of recursive domain equations, *Siam Journal of Computing* vol. 11 no. 4, 1982.
- [63] J. C. Reynolds: Polymorphism is not set-theoretic, Symposium on Semantics of Data Types (ed. Kahn, McQueen, Plotkin), Springer LNCS 173, 1984.
- [64] E. P. Robinson: Logical aspects of denotational semantics, in [60].
- [65] E. P. Robinson: How complete is PER? in: Proc. of 4th Annual Symposium on Logic in Computer Science, 1989.
- [66] E. P. Robinson, G. Rosolini: Colimit completions and the effective topos, to appear.
- [67] H. Rogers, Jr.: *Theory of Recursive Functions and Effective Computability*, McGraw Hill, 1967.
- [68] G. Rosolini: *Continuity and Effectiveness in Topoi*, D. Phil. thesis (Oxford; Carnegie-Mellon), 1986.

- [69] G. Rosolini: Categories and effective computations, in [60].
- [70] G. Rosolini: Uniformity, dinaturality and 2.
- [71] G. Rosolini: An Exper model for Quest, to appear in the proceedings of MFPS91.
- [72] G. Rosolini: About modest sets, to appear.
- [73] D.S. Scott: Data types as lattices, *SIAM J. Comput.* 5 (3) (1976) 522-587.
- [74] D.S. Scott: unpublished notes written at Schwangau, 1989.
- [75] T. Streicher: Independence results for calculi of dependent types, in: *Lecture Notes in Computer Science* 389, Springer, 1989.
- [76] P. Taylor: *Recursive Domains, Indexed Category Theory and Polymorphism*, Ph.D. thesis (Cambridge), 1986.
- [77] P. Taylor: The fixed point property in synthetic domain theory, to appear in the proceedings of the 6th Annual Symposium on Logic in Computer Science (Amsterdam, 1991).
- [78] S.J. Vickers: *Topology via Logic*, Cambridge Tracts in Theoretical Computer Science 5, CUP, 1989.
- [79] M. Zamfir: Initial algebra semantics and concurrency, in: *Mathematical Foundations of Programming Language Semantics*, *Lecture Notes in Computer Science* 298, Springer, 1988.

Index

- A, *see* partial combinatory algebra
- A (object of topos), 32
- A-valued set, 31
- algebraically compact category, 199
- algebraicity, 136
- assembly, 27
 - global sections of, 28
- axiom of choice
 - countable, 33
- axiomatic domain theory
 - Hyland's, 199
 - Taylor's, 200
- Böhm tree
 - equality in graph model, 167
 - intersection, 185
- bounded quantification, 60
- canonically focal
 - in Böhm tree model, 182
 - in graph model, 170
- $\mathcal{CFSig}_\perp$, 111
- chain completion, 87
- chains
 - in Σ -spaces, 84, 91
 - in Böhm tree model, 183
 - in graph model, 172
- Church's thesis, 33
 - unexpected failure of, 36
- coercion, 59
- complete PER
 - in Böhm tree model, 183
- complete Σ -space, 84
- completeness
 - different kinds of, 37
- computable partial maps, 103
 - category of, 113
- constructions, theory of
 - PER model, 30
- continuity
 - for complete Σ -spaces, 85
 - in Böhm tree model, 183
 - in graph model, 172
- continuously dependent type, 152
- contra-co rule, 60, 62
- convergence predicate, 104
- convex powerdomain, 123
- $\mathcal{CSig}$, 84
- Δ -space, 89
- dependent type
 - as sheaf, 152
- display map, 120
- distributive lattice, 129
- domain equations, 199
- domain theory
 - axiomatic, 11
 - classical, 10
 - domains as PERs, 11, 21, 22
 - domains as sets, 11, 202

- synthetic, 11, 199
- dominance, 72
 - relative, 167
- double negation topology, 32
 - closed subobjects, 34
- effective topos, 12, 32
 - internal logic of, 33
 - pros and cons, 20
- Egli-Milner order, 124
- extensional PER, 195
- externalization
 - of an internal category, 39
 - stack completion, 39
- fibrations
 - and polymorphism, 57
- fixed point category, 102, 192
- fixed point combinator
 - approximations to, 183
 - calculates fixed points, 185
- fixed point property, 192, 199
- fixed points
 - in PONDER, 22
- flat morphism
 - would like a notion of, 164
- flat Σ -space, 89
- Floyd-Hoare logic
 - Σ_1^0 fragment of, 131
- focal
 - in graph model, 172
- focal object, 101
- $F\omega$, 57
- free algebra (for lift monad), 113
- generic chain
 - for linked Σ -spaces, 91
 - in Böhm tree model, 184
 - in graph model, 172
- generic chain with a sup
 - for complete Σ -spaces, 91
 - in Böhm tree model, 184
 - in graph model, 173
- generic chain with a sup and upper
 - bound, in graph model, 173
- generic type, 58
- graph model
 - general construction, 166
- gros toposes, 160
 - and polymorphism, 162
- groupoid interpretation, 47
 - parametricity, 51
- Hoare order, 129
- Hoare powerdomain, 129
- inclusion, 55
 - in fibration, 61
 - in PER model, 66
 - not invariant under isomorphism, 56
- inclusions
 - in reflective submodels, 68
- inclusive predicate, 154
- indexed products
 - closure under, 165
- inheritance, 55
- injection-projection pair, 94
- join-semilattice, 129
- K_1 , 25
- Kleene's first model, 25

- KLS property, 144
- K^+X (finite nonempty subsets of X), 124
- Kristin English, 3
- KX (finite subsets), 126
- $\mathcal{L}$, 75
- LAMBDA, 167
- λ -calculus
 - second-order, 26
- lax limit, 97
- least fixed point
 - generic, 92
 - in graph model, 175
- lift construction, 104
 - applied to algebras, 128
 - as a monad, 111
 - in Böhm tree model, 182
 - in graph model, 169
 - recursion-theoretic definition, 107
- limit-colimit coincidence, 94
- link relation, 81
 - in graph model, 170
- link-ordered, 82
- link-preordered, 81
- linked Σ -space, 78
- local homeomorphism, 152
- logical point, 138
- logically open, 133
- lower powerdomain, 129
- $\mathcal{LSig}$, 79
- m -reducibility, 37
- Markov's principle, 33
 - modified realizability, 35
- meet relation, 186
- meet-semilattice, 129
- mixed powerdomain, 130
- Mod
 - completeness of, 29
 - subcategory of Sep , 28
 - subfibration of Sep , 29
- Mod (internal category)
 - completeness of, 36, 37
 - in Sep , 29
 - in realizability topos, 36
- modal operators
 - for convex powerdomain, 126
 - for other powerdomains, 129
 - for strictified algebras, 128
- modest set
 - in Sep , 28
 - in realizability topos, 37
- Moggi-Hyland interpretation
 - problem with, 46
- ∇ , 28
- $NM(X)$ (nondeterministic multivalued domain), 128
- $\mathbb{N}^{\mathbb{N}}$
 - product topology, 143
- nondeterministic multivalued computations, 128
- $\dashv\dashv$ -closed subobjects
 - classified by $\nabla 2$, 34
 - correspond to subsets, 34
- object-oriented programming, 55
- ω (generic chain)
 - for linked Σ -spaces, 91
- $\bar{\omega}$ (generic chain with sup)
 - for complete Σ -spaces, 91

- open map or object, 133
- Ord*, 142
- $\text{Orth}(\nabla^2)$, 40, 117
- $\text{Orth}(\Sigma)$, 89
- orthogonal, 75
- packages, 157
- parallelism
 - and sheaves, 148
- parametricity, 50
- partial combinatory algebra, 24
- partial equivalence relation, 25
- PCA, 167
- PCF, 167
- PER, 25
 - Abadi/Plotkin definitions, 197
 - extensional, 195
- Per, 25
 - embedding in *Sep*, 28
- PER models, 24
- Plotkin powerdomain, 123
- point
 - logical, 138
 - topological, 137
- polymorphic λ -calculus, 26
- polymorphism, 26
 - and fibrations, 57
 - groupoid interpretation, 47
 - Moggi-Hyland interpretation, 30, 46
 - parametricity, 50, 53
 - PER model, 26
 - stability under equivalence, 46, 48
 - uniformized interpretation, 52
- powerdomain
 - convex or Plotkin, 123
 - lower or Hoare, 129
 - mixed, 130
 - sandwiches, 130
 - upper or Smyth, 129
- powerkinds, 60, 63
 - in PER model, 66
- predicates
 - sound, 201
- process algebra
 - completely trivial, 128
- projection, 121
- projection (strictness analysis), 115
- Quest, 57
- R (graph model), 167
- r.e. graph model, 167
- r.e. set
 - extensional, 143
 - with KLS property, 144
- r.e. subobject classifier, 71
- realizability topos
 - constructions, 30
- realizability toposes, 12
- recursive mathematics, 33
- reflectivity, 40
 - criterion for, 179
 - of fibrations, 40
 - of modest sets, 40
 - of subcategories, 41
 - weak, 40
- regular subobjects
 - in *Sep*, 34
- relative computability, 41
- relative dominance, 167

- Rep*, 141
- replete object, 140
- sandwiches powerdomain, 130
- semilattice, 123
- Sep*, 27
 - completeness of, 28
 - embedding in realizability topos, 31
- Sets*
 - embedding in *Sep*, 28
- $\text{Sh}(X)$, 147
- sheaf, 147
 - of sections, 149
- Sig*, 73
- sig** (internal category), 73
- Σ
 - in Böhm tree model, 180
 - in graph model, 168
- Σ (r.e. subobject classifier), 71
- Σ -anodyne, 140
- σ -domain, 194
- Σ -fibred family, 118
- Σ -injective, 111
- Σ -order, 74
 - pointwise, 77
- Σ -partial map classifier, 104
- σ -set, 193
- Σ -space, 73
 - complete, 84
 - flat, 89
 - focal, 101
 - linked, 78
- Σ -subset, 71
- Σ -topology, 137
- Smyth order, 129
- Smyth powerdomain, 129
- sober Σ -space, 137, 138
 - Rosolini's notion of, 193
- sound predicates, 201
- specifications
 - as sheaves, 157
- stability, 186
- stack completion, 39
- stalk space, 149, 151
- strict maps
 - category of, 111
- subkinds, 60, 64
- subspace, 75
- substitution functor, 58
- subsumption, 56, 59
- subtyping, 55
 - and constructions on types, 69
 - in PER models, 65
 - in reflective submodels, 67
- topological point, 137
- total correctness, 132
- Turing degrees, 41
 - and orthogonality, 44
- type inclusion, 55
- uniformized intrerpretation, 52
- uniformly complete, 173
- upper powerdomain, 129
- versality, principle of, 199
- weakly reflective, 40
- weighted limit, 99
- ω -Ord, 142
- $(X_i)_{i \in I}$ (I -indexed family), 117

$Z_A, 37$